FORTINET

FortiOS™ Handbook - IPsec VPN

VERSION 6.0.2

FORTINET DOCUMENT LIBRARY

https://docs.fortinet.com

FORTINET VIDEO GUIDE

https://video.fortinet.com

FORTINET KNOWLEDGE BASE

http://kb.fortinet.com

FORTINET BLOG

https://blog.fortinet.com

CUSTOMER SERVICE & SUPPORT

https://support.fortinet.com

http://cookbook.fortinet.com/how-to-work-with-fortinet-support/

FORTINET COOKBOOK

http://cookbook.fortinet.com

FORTINET TRAINING AND CERTIFICATION PROGRAM

https://www.fortinet.com/support-and-training/training.html

NSE INSTITUTE

https://training.fortinet.com/

FORTIGUARD CENTER

https://fortiguard.com

FORTICAST

http://forticast.fortinet.com

END USER LICENSE AGREEMENT

https://www.fortinet.com/doc/legal/EULA.pdf

July 26, 2018

FortiOS™ Handbook - IPsec VPN

01-602-481080-20180726

TABLE OF CONTENTS

Change log

Date	Change description
July 26, 2018	FortiOS 6.0.2 document release. See "What's new in FortiOS 6.0.2" on page 10.
June 5, 2018	FortiOS 6.0.1 document release. See "What's new in FortiOS 6.0.1" on page 10.
March 29, 2018	FortiOS 6.0 document release. See "What's new in FortiOS 6.0" on page 10.

Introduction

This FortiOS Handbook chapter contains the following sections:

IPsec VPN concepts explains the basic concepts that you need to understand about virtual private networks (VPNs).

IPsec VPN overview provides a brief overview of IPsec technology and includes general information about how to configure IPsec VPNs using this guide.

IPsec VPN in the web-based manager describes the IPsec VPN menu of the web-based manager interface.

Gateway-to-gateway configurations explains how to set up a basic gateway-to-gateway (site-to-site) IPsec VPN. In a gateway-to-gateway configuration, two FortiGate units create a VPN tunnel between two separate private networks.

Hub-and-spoke configurations describes how to set up hub-and-spoke IPsec VPNs. In a hub-and-spoke configuration, connections to a number of remote peers and/or clients radiate from a single, central FortiGate hub.

Dynamic DNS configuration describes how to configure a site-to-site VPN, in which one FortiGate unit has a static IP address and the other FortiGate unit has a dynamic IP address and a domain name.

FortiClient dialup-client configurations guides you through configuring a FortiClient dialup-client IPsec VPN. In a FortiClient dialup-client configuration, the FortiGate unit acts as a dialup server and VPN client functionality is provided by the FortiClient Endpoint Security application installed on a remote host.

FortiGate dialup-client configurations explains how to set up a FortiGate dialup-client IPsec VPN. In a FortiGate dialup-client configuration, a FortiGate unit with a static IP address acts as a dialup server and a FortiGate unit with a dynamic IP address initiates a VPN tunnel with the FortiGate dialup server.

Supporting IKE Mode config clients explains how to set up a FortiGate unit as either an IKE Mode Config server or client. IKE Mode Config is an alternative to DHCP over IPsec.

Internet-browsing configuration explains how to support secure web browsing performed by dialup VPN clients, and hosts behind a remote VPN peer. Remote users can access the private network behind the local FortiGate unit and browse the Internet securely. All traffic generated remotely is subject to the security policy that controls traffic on the private network behind the local FortiGate unit.

Redundant VPN configurations discusses the options for supporting redundant and partially redundant tunnels in an IPsec VPN configuration. A FortiGate unit can be configured to support redundant tunnels to the same remote peer if the FortiGate unit has more than one interface to the Internet.

Transparent mode VPNs describes two FortiGate units that create a VPN tunnel between two separate private networks transparently. In transparent mode, all FortiGate unit interfaces except the management interface are invisible at the network layer.

IPv6 IPsec VPNs describes FortiGate unit VPN capabilities for networks based on IPv6 addressing. This includes IPv4-over-IPv6 and IPv6-over-IPv4 tunnelling configurations. IPv6 IPsec VPNs are available in FortiOS 3.0 MR5 and later.

L2TP and IPsec (Microsoft VPN) explains how to support Microsoft Windows native VPN clients.

GRE over IPsec (Cisco VPN) explains how to interoperate with Cisco VPNs that use Generic Routing Encapsulation (GRE) protocol with IPsec.

Protecting OSPF with IPsec provides an example of protecting OSPF links with IPsec.

Redundant OSPF routing over IPsec provides an example of redundant secure communication between two remote networks using an OSPF VPN connection.

OSPF over dynamic IPsec provides an example of how to create a dynamic IPsec VPN tunnel that allows OSPF.

BGP over dynamic IPsec provides an example of how to create a dynamic IPsec VPN tunnel that allows BGP.

Phase 1 parameters provides detailed step-by-step procedures for configuring a FortiGate unit to accept a connection from a remote peer or dialup client. The basic Phase 1 parameters identify the remote peer or clients and support authentication through preshared keys or digital certificates. You can increase VPN connection security further using methods such as extended authentication (XAuth).

Phase 2 parameters provides detailed step-by-step procedures for configuring an IPsec VPN tunnel. During Phase 2, the specific IPsec security associations needed to implement security services are selected and a tunnel is established.

Defining VPN security policies explains how to specify the source and destination IP addresses of traffic transmitted through an IPsec VPN tunnel, and how to define a security encryption policy. Security policies control all IP traffic passing between a source address and a destination address.

Logging and monitoring and Troubleshooting provide VPN monitoring and troubleshooting procedures.

What's new in FortiOS 6.0.2

The following list contains new IPsec VPN features added in FortiOS 6.0.2. Click on a link to navigate to that section for further information.

- "OCVPN support for High Availability (HA)" on page 111

What's new in FortiOS 6.0.1

The following list contains new IPsec VPN features added in FortiOS 6.0.1. Click on a link to navigate to that section for further information.

- Updates to "IPsec support for ChaCha20/Poly1305 AEAD cipher" on page 71
- "IPsec support for AES-GCM for IKEv2 Phase 1" on page 71

What's new in FortiOS 6.0

The following list contains new IPsec VPN features added in FortiOS 6.0. Click on a link to navigate to that section for further information.

- Updates to "IPsec VPN Wizard options" on page 13
- "Curve25519 128-bit elliptic curve group" on page 18

IPsec VPN concepts

Virtual Private Network (VPN) technology enables remote users to connect to private computer networks to gain access to their resources in a secure way. For example, an employee traveling or working from home can use a VPN to securely access the office network through the Internet.

Instead of remotely logging on to a private network using an unencrypted and unsecure Internet connection, the use of a VPN ensures that unauthorized parties cannot access the office network and cannot intercept any of the information that is exchanged between the employee and the office. It is also common to use a VPN to connect the private networks of two or more offices.

Fortinet offers VPN capabilities in the FortiGate Unified Threat Management (UTM) appliance and in the FortiClient Endpoint Security suite of applications. A FortiGate unit can be installed on a private network, and FortiClient software can be installed on the user's computer. It is also possible to use a FortiGate unit to connect to the private network instead of using FortiClient software.

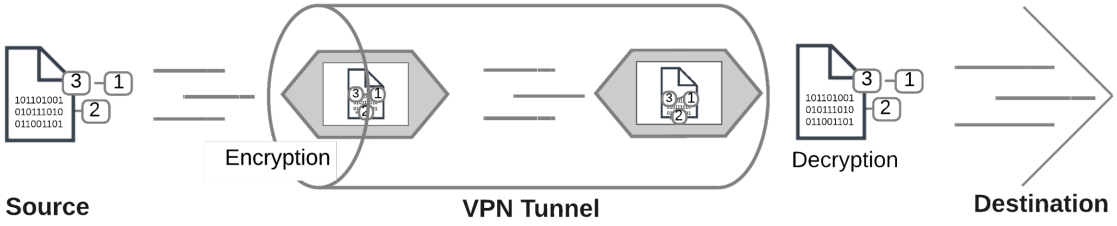
This chapter discusses VPN terms and concepts including:

VPN tunnels

The data path between a user's computer and a private network through a VPN is referred to as a tunnel. Like a physical tunnel, the data path is accessible only at both ends. In the telecommuting scenario, the tunnel runs between the FortiClient application on the user's PC, or a FortiGate unit or other network device and the FortiGate unit on the office private network.

Encapsulation makes this possible. IPsec packets pass from one end of the tunnel to the other and contain data packets that are exchanged between the local user and the remote private network. Encryption of the data packets ensures that any third-party who intercepts the IPsec packets can not access the data.

Encoded data going through a VPN tunnel



You can create a VPN tunnel between:

- A PC equipped with the FortiClient application and a FortiGate unit
- Two FortiGate units
- Third-party VPN software and a FortiGate unit
  For more information on third-party VPN software, refer to the Fortinet Knowledge Base for more information.

Tunnel templates

Several tunnel templates are available in the IPsec VPN Wizard that cover a variety of different types of IPsec VPN. A list of these templates appear on the first page of the Wizard, located at **VPN > IPsec Wizard**. The tunnel template list follows.

IPsec VPN Wizard options

VPN Type	Remote Device Type	NAT Options	Description
Site to Site	**FortiGate**	• No NAT between sites • This site is behind NAT • The remote site is behind NAT	Static tunnel between this FortiGate and a remote FortiGate.
	Cisco	• No NAT between sites • This site is behind NAT • The remote site is behind NAT	Static tunnel between this FortiGate and a remote Cisco firewall.

VPN Type	Remote Device Type		NAT Options	Description
Remote Access	**Client-based**	**FortiClient VPN for OS X, Windows, and Android**	N/A	On-demand tunnel for users using the FortiClient software.
		Cisco AnyConnect	N/A	On-demand tunnel for users using the Cisco IPsec client.
	Native	**iOS Native**	N/A	On-demand tunnel for iPhone/iPad users using the native iOS IPsec client.
		Android Native	N/A	On-demand tunnel for Android users using the native L2TP/IPsec client.
		Windows Native	N/A	On-demand tunnel for Android users using the native L2TP/IPsec client.
Custom	N/A		N/A	No Template.

Cisco's VPN Client has reached its End-of-Life/End-of-Support as of July 30, 2016, and has been replaced by Cisco AnyConnect Secure Mobility Client.

In FortiOS 5.6.4+, the first step of the VPN Creation Wizard (**VPN > IPsec Wizard**) delineates the **Remote Device Type** (for **Remote Access** templates) between **Client-based** and **Native** in order to distinguish FortiClient and Cisco device options from native OS device options.

VPN tunnel list

Once you create an IPsec VPN tunnel, it appears in the VPN tunnel list at **VPN > IPsec Tunnels**. By default, the tunnel list indicates the name of the tunnel, its interface binding, the tunnel template used, and the tunnel status. If you right-click on the table header row, you can include columns for comments, IKE version, mode (aggressive vs main), phase 2 proposals, and reference number. The tunnel list page also includes the option to create a new tunnel, as well as the options to edit or delete a highlighted tunnel.

FortiView VPN tunnel map

A geospatial map can be found under **FortiView > VPN Map** to help visualize IPsec (and SSL) VPN connections to a FortiGate using Google Maps. This feature adds a geographical-IP API service for resolving spatial locations from IP addresses.

VPN gateways

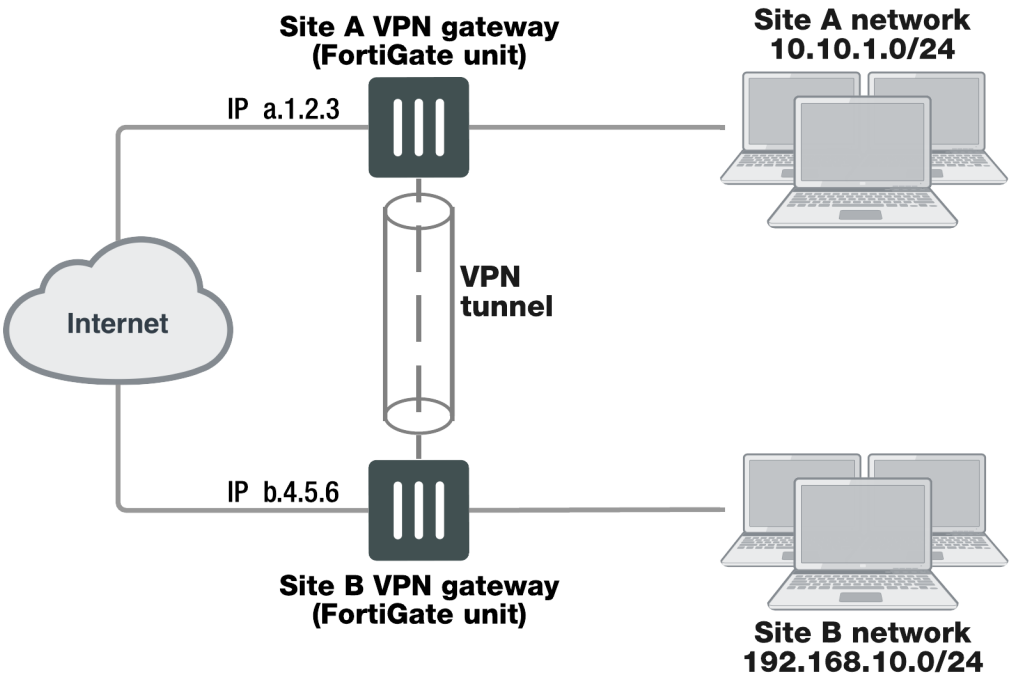
A gateway is a router that connects the local network to other networks. The default gateway setting in your computer's TCP/IP properties specifies the gateway for your local network.

A VPN gateway functions as one end of a VPN tunnel. It receives incoming IPsec packets, decrypts the encapsulated data packets and passes the data packets to the local network. Also, it encrypts data packets destined for the other end of the VPN tunnel, encapsulates them, and sends the IPsec packets to the other VPN gateway. The VPN gateway is a FortiGate unit because the private network behind it is protected, ensuring the security of the unencrypted VPN data. The gateway can also be FortiClient software running on a PC since the unencrypted data is secure on the PC.

The IP address of a VPN gateway is usually the IP address of the network interface that connects to the Internet. Optionally, you can define a secondary IP address for the interface and use that address as the local VPN gateway address. The benefit of doing this is that your existing setup is not affected by the VPN settings.

The following diagram shows a VPN connection between two private networks with FortiGate units acting as the VPN gateways. This configuration is commonly referred to as Gateway-to-Gateway IPsec VPN.

VPN tunnel between two private networks



Although the IPsec traffic may actually pass through many Internet routers, you can visualize the VPN tunnel as a simple secure connection between the two FortiGate units.

Users on the two private networks do not need to be aware of the VPN tunnel. The applications on their computers generate packets with the appropriate source and destination addresses, as they normally do. The FortiGate units manage all the details of encrypting, encapsulating, and sending the packets to the remote VPN gateway.

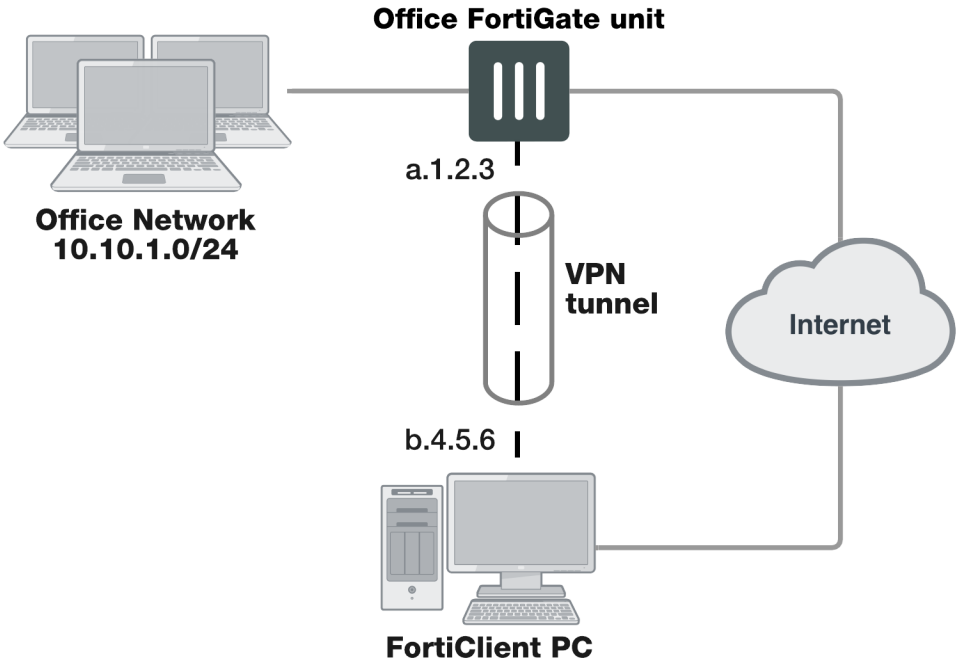
The data is encapsulated in IPsec packets only in the VPN tunnel between the two VPN gateways. Between the user's computer and the gateway, the data is on the secure private network and it is in regular IP packets.

For example User1 on the Site A network, at IP address 10.10.1.7, sends packets with destination IP address 192.168.10.8, the address of User2 on the Site B network. The Site A FortiGate unit is configured to send packets with destinations on the 192.168.10.0 network through the VPN, encrypted and encapsulated. Similarly, the Site B FortiGate unit is configured to send packets with destinations on the 10.10.1.0 network through the VPN tunnel to the Site A VPN gateway.

In the site-to-site, or gateway-to-gateway VPN shown below, the FortiGate units have static (fixed) IP addresses and either unit can initiate communication.

You can also create a VPN tunnel between an individual PC running FortiClient and a FortiGate unit, as shown below. This is commonly referred to as Client-to-Gateway IPsec VPN.

VPN tunnel between a FortiClient PC and a FortiGate unit



On the PC, the FortiClient application acts as the local VPN gateway. Packets destined for the office network are encrypted, encapsulated into IPsec packets, and sent through the VPN tunnel to the FortiGate unit. Packets for other destinations are routed to the Internet as usual. IPsec packets arriving through the tunnel are decrypted to recover the original IP packets.

Clients, servers, and peers

A FortiGate unit in a VPN can have one of the following roles:

- **Server** — responds to a request to establish a VPN tunnel.
- **Client** — contacts a remote VPN gateway and requests a VPN tunnel.
- **Peer** — brings up a VPN tunnel or responds to a request to do so.

The site-to-site VPN shown above is a peer-to-peer relationship. Either FortiGate unit VPN gateway can establish the tunnel and initiate communications. The FortiClient-to-FortiGate VPN shown below is a client-server relationship. The FortiGate unit establishes a tunnel when the FortiClient PC requests one.

A FortiGate unit cannot be a VPN server if it has a dynamically-assigned IP address. VPN clients need to be configured with a static IP address for the server. A FortiGate unit acts as a server only when the remote VPN gateway has a dynamic IP address or is a client-only device or application, such as FortiClient.

As a VPN server, a FortiGate unit can also offer automatic configuration for FortiClient PCs. The user needs to know only the IP address of the FortiGate VPN server and a valid user name/password. FortiClient downloads the VPN configuration settings from the FortiGate VPN server. For information about configuring a FortiGate unit as a VPN server, see the FortiClient Administration Guide.

Encryption

Encryption mathematically transforms data to appear as meaningless random numbers. The original data is called plaintext and the encrypted data is called ciphertext. The opposite process, called decryption, performs the inverse operation to recover the original plaintext from the ciphertext.

The process by which the plaintext is transformed to ciphertext and back again is called an algorithm. All algorithms use a small piece of information, a key, in the arithmetic process of converted plaintext to ciphertext, or vice-versa. IPsec uses symmetrical algorithms, in which the same key is used to both encrypt and decrypt the data. The security of an encryption algorithm is determined by the length of the key that it uses. FortiGate IPsec VPNs offer the following encryption algorithms, in descending order of security:

Encryption	Description
ChaCha20/Poly1305	A combination of the ChaCha20 symmetric cipher and Poly1305-AES, a variant of the AES 128-bit block algorithm that uses a 128-bit key and an 128-bit nonce.
AES-GCM	Galois/Counter Mode (GCM), a block cipher mode of operation providing both confidentiality and data origin authentication.
AES256	A 128-bit block algorithm that uses a 256-bit key.
AES192	A 128-bit block algorithm that uses a 192-bit key.
AES128	A 128-bit block algorithm that uses a 128-bit key.

Encryption	Description
3DES	Triple-DES, in which plain text is DES-encrypted three times by three keys.
DES	Digital Encryption Standard, a 64-bit block algorithm that uses a 56-bit key

The default encryption algorithms provided on FortiGate units make recovery of encrypted data almost impossible without the proper encryption keys.

There is a human factor in the security of encryption. The key must be kept secret, known only to the sender and receiver of the messages. Also, the key must not be something that unauthorized parties might easily guess, such as the sender's name, birthday or simple sequence such as `123456`.

Diffie-Hellman groups

FortiOS IPsec VPN supports the following Diffie-Hellman (DH) asymmetric key algorithms for public key cryptography.

DH Group	Description
1	More Modular Exponential (MODP) DH Group with a 768-bit modulus
2	MODP with a 1024-bit modulus
5	MODP with a 1536-bit modulus
14	MODP with a 2048-bit modulus
15	MODP with a 3027-bit modulus
16	MODP with a 4096-bit modulus
17	MODP with a 6144-bit modulus
18	MODP with a 8192-bit modulus
19	256-bit random elliptic curve group
20	384-bit random elliptic curve group
21	521-bit random elliptic curve group
27	Brainpool 224-bit elliptic curve group
28	Brainpool 256-bit elliptic curve group
29	Brainpool 384-bit elliptic curve group
30	Brainpool 512-bit elliptic curve group
31	Curve25519 128-bit elliptic curve group

* When using aggressive mode, DH groups cannot be negotiated.

By default, DH group 14 is selected, to provide sufficient protection for stronger cipher suites that include AES and SHA2. If you select multiple DH groups, the order they appear in the configuration is the order in which they are negotiates.

If both VPN peers (or a VPN server and its client) have static IP addresses and use aggressive mode, select a single DH group. The setting on the FortiGate unit must be identical to the setting on the remote peer or dialup client.

When the remote VPN peer or client has a dynamic IP address and uses aggressive mode, select up to three DH groups on the FortiGate unit and one DH group on the remote peer or dialup client. The setting on the remote peer or dialup client must be identical to one of the selections on the FortiGate unit.

If the VPN peer or client employs main mode, you can select multiple DH groups. At least one of the settings on the remote peer or dialup client must be identical to the selections on the FortiGate unit.

IPsec overheads

The FortiGate sets an IPsec tunnel Maximum Transmission Unit (MTU) of 1436 for 3DES/SHA1 and an MTU of 1412 for AES128/SHA1, as seen with `diag vpn tunnel list`. This indicates that the FortiGate allocates 64 bytes of overhead for 3DES/SHA1 and 88 bytes for AES128/SHA1, which is the difference if you subtract this MTU from a typical ethernet MTU of 1500 bytes.

During the encryption process, AES/DES operates using a specific size of data which is block size. If data is smaller than that, it will be padded for the operation. MD5/SHA-1 HMAC also operates using a specific block size.

The following table describes the potential maximum overhead for each IPsec encryption:

IPsec Transform Set	IPsec Overhead (Max. bytes)
ESP-AES (256, 192, or 128),ESP-SHA-HMAC, or MD5	88
ESP-AES (256, 192, or 128)	61
ESP-3DES, ESP-DES	45
ESP-(DES or 3DES), ESP-SHA-HMAC, or MD5	64
ESP-Null, ESP-SHA-HMAC, or MD5	45
AH-SHA-HMAC or MD5	44

Authentication

To protect data via encryption, a VPN must ensure that only authorized users can access the private network. You must use either a preshared key on both VPN gateways or RSA X.509 security certificates. The examples in this guide use only preshared key authentication. Refer to the Fortinet Knowledge Base for articles on RSA X.509 security certificates.

Preshared keys

A preshared key contains at least six random alphanumeric characters. Users of the VPN must obtain the preshared key from the person who manages the VPN server and add the preshared key to their VPN client configuration.

Although it looks like a password, the preshared key, also known as a shared secret, is never sent by either gateway. The preshared key is used in the calculations at each end that generate the encryption keys. As soon as the VPN peers attempt to exchange encrypted data, preshared keys that do not match will cause the process to fail.

Additional authentication

To increase security, you can require additional means of authentication from users, such as:

- An identifier, called a peer ID or a local ID.
- Extended authentication (XAUTH) which imposes an additional user name/password requirement.

A Local ID is an alphanumeric value assigned in the Phase 1 configuration. The Local ID of a peer is called a Peer ID.

In FortiOS 5.2, new authentication methods have been implemented for IKE: ECDSA-256, ECDSA-384, and ECDSA-521. However, AES-XCBC is not supported.

Full CA chain checking

Added a new option (enabled by default) to fail certificate verification if any of the CAs in the trust chain are not found in the CA store. When disabled, a sub-CA is sufficient to pass certificate verification.

Syntax

```
config vpn certificate setting
   set check-ca-chain {enable | disable}
end
```

Phase 1 and Phase 2 settings

A VPN tunnel is established in two phases: Phase 1 and Phase 2. Several parameters determine how this is done. Except for IP addresses, the settings simply need to match at both VPN gateways. There are defaults that are appropriate for most cases.

FortiClient distinguishes between Phase 1 and Phase 2 only in the VPN Advanced settings and uses different terms. Phase 1 is called the IKE Policy. Phase 2 is called the IPsec Policy.

Phase 1

In Phase 1, the two VPN gateways exchange information about the encryption algorithms that they support and then establish a temporary secure connection to exchange authentication information.

When you configure your FortiGate unit or FortiClient application, you must specify the following settings for Phase 1:

Remote gateway	The remote VPN gateway's address. FortiGate units also have the option of operating only as a server by selecting the "Dialup User" option.
Preshared key	This must be the same at both ends. It is used to encrypt Phase 1 authentication information.
Local interface	The network interface that connects to the other VPN gateway. This applies on a FortiGate unit only.

All other Phase 1 settings have default values. These settings mainly configure the types of encryption to be used. The default settings on FortiGate units and in the FortiClient application are compatible. The examples in this guide use these defaults.

For more detailed information about Phase 1 settings, see Phase 1 parameters on page 46.

Phase 2

Similar to the Phase 1 process, the two VPN gateways exchange information about the encryption algorithms that they support for Phase 2. You may choose different encryption for Phase 1 and Phase 2. If both gateways have at least one encryption algorithm in common, a VPN tunnel can be established. Keep in mind that more algorithms each phase does not share with the other gateway, the longer negotiations will take. In extreme cases this may cause timeouts during negotiations.

To configure default Phase 2 settings on a FortiGate unit, you need only select the name of the corresponding Phase 1 configuration. In FortiClient, no action is required to enable default Phase 2 settings.

For more detailed information about Phase 2 settings, see Phase 2 parameters on page 66.

Security Association

The establishment of a Security Association (SA) is the successful outcome of Phase 1 negotiations. Each peer maintains a database of information about VPN connections. The information in each SA can include cryptographic algorithms and keys, keylife, and the current packet sequence number. This information is kept synchronized as the VPN operates. Each SA has a Security Parameter Index (SPI) that is provided to the remote peer at the time the SA is established. Subsequent IPsec packets from the peer always reference the relevant SPI. It is possible for peers to have multiple VPNs active simultaneously, and correspondingly multiple SPIs.

The IPsec SA connect message generated is used to install dynamic selectors. These selectors can be installed via the auto-negotiate mechanism. When phase 2 has auto-negotiate enabled, and phase 1 has mesh selector-type set to **subnet**, a new dynamic selector will be installed for each combination of source and destination subnets. Each dynamic selector will inherit the auto-negotiate option from the template selector and begin SA negotiation. Phase 2 selector sources from dial-up clients will all establish SAs without traffic being initiated from the client subnets to the hub.

Remote IP address change detection

SAs are stored in a hash table when keyed off the IPsec SA SPI value. This enables the FortiGate, for each inbound ESP packet received, to immediately look up the SA and compare the stored IP address against the one in the incoming packet. If the incoming and stored IP addresses differ, an IP address change can be made in the kernel SA, and an update event can be triggered for IKE.

IKE and IPsec packet processing

Internet Key Exchange (IKE) is the protocol used to set up SAs in IPsec negotiation. As described in Phase 1 parameters on page 46, you can optionally choose IKEv2 over IKEv1 if you configure a route-based IPsec VPN. IKEv2 simplifies the negotiation process, in that it provides no choice of Aggressive or Main mode in Phase 1. IKEv2 also uses less bandwidth.

The following sections identify how IKE versions 1 and 2 operate and differentiate.

IKEv1

Phase 1

A peer, identified in the IPsec policy configuration, begins the IKE negotiation process. This IKE Security Association (SA) agreement is known as Phase 1. The Phase 1 parameters identify the remote peer or clients and supports authentication through pre-shared key (PSK) or digital certificate. You can increase access security further using peer identifiers, certificate distinguished names, group names, or the FortiGate extended authentication (XAuth) option for authentication purposes. Basically, Phase 1 authenticates a remote peer and sets up a secure communication channel for establishing Phase 2, which negotiates the IPsec SA.

IKE Phase 1 can occur in either Main mode or Aggressive mode. For more information, see Phase 1 parameters on page 46.

IKE Phase 1 is successful only when the following are true:

- Each peer negotiates a matching IKE SA policy.
- Each peer is authenticated and their identities protected.
- The Diffie-Hellman exchange is authenticated (the pre-shared secret keys match).

For more information on Phase 1, see Phase 1 parameters on page 46.

Phase 2

Phase 2 parameters define the algorithms that the FortiGate unit can use to encrypt and transfer data for the remainder of the session in an IPsec SA. The basic Phase 2 settings associate IPsec Phase 2 parameters with a Phase 1 configuration.

In Phase 2, the VPN peer or client and the FortiGate unit exchange keys again to establish a more secure communication channel. The Phase 2 Proposal parameters select the encryption and authentication algorithms needed to generate keys for protecting the implementation details of the SA. The keys are generated automatically using a Diffie-Hellman algorithm.

In Phase 2, Quick mode selectors determine which IP addresses can perform IKE negotiations to establish a tunnel. By only allowing authorized IP addresses access to the VPN tunnel, the network is more secure. For more information, see Phase 2 parameters on page 66.

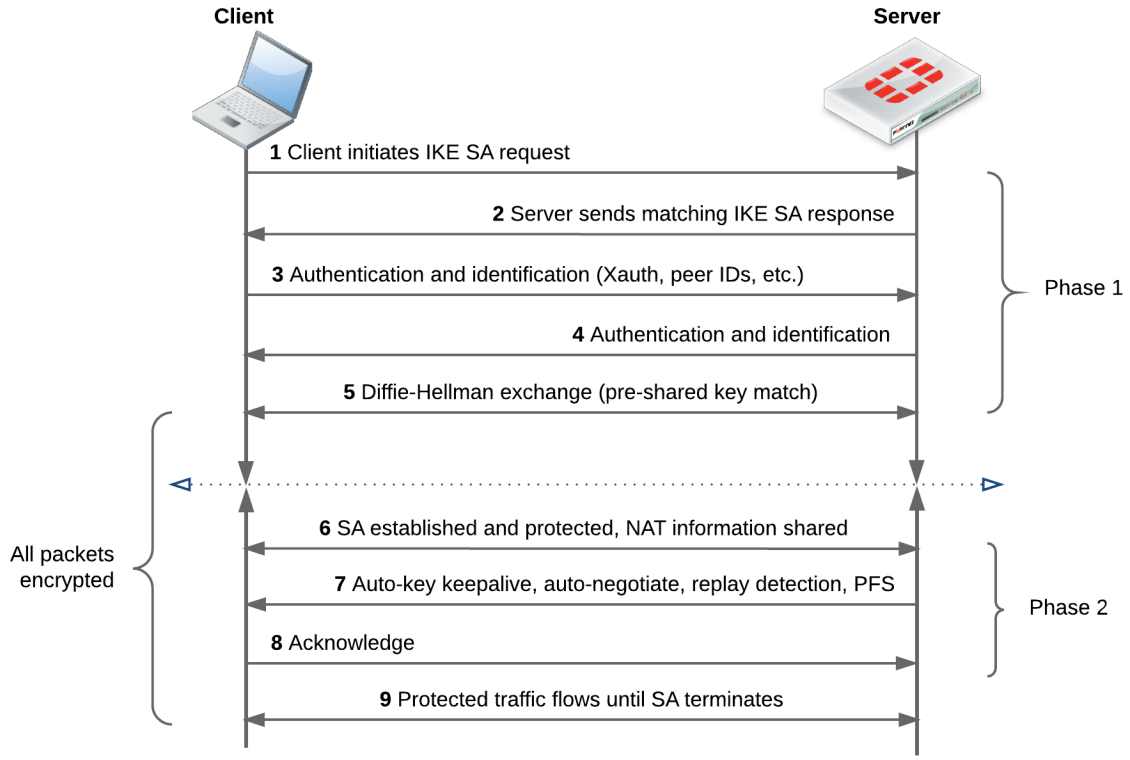
IKE Phase 2 is successful only when the following are true:

- The IPsec SA is established and protected by the IKE SA.
- The IPsec SA is configured to renegotiate after set durations (see Phase 2 parameters on page 66 and Phase 2 parameters on page 66).
- **Optional:** Replay Detection is enabled. Replay attacks occur when an unauthorized party intercepts a series of IPsec packets and replays them back into the tunnel. See Phase 2 parameters on page 66.
- **Optional:** Perfect Forward Secrecy (PFS) is enabled. PFS improves security by forcing a new Diffie-Hellman exchange whenever keylife expires. See Phase 2 parameters on page 66.

For more information on Phase 2, see Phase 2 parameters on page 66.

With Phase 2 established, the IPsec tunnel is fully negotiated and traffic between the peers is allowed until the SA terminates (for any number of reasons; time-out, interruption, disconnection, etc).

The entire IKEv1 process is demonstrated in the following diagram:



IKEv2

Phase 1

Unlike Phase 1 of IKEv1, IKEv2 does not provide options for Aggressive or Main mode. Furthermore, Phase 1 of IKEv2 begins immediately with an IKE SA initiation, consisting of only two packets (containing all the information typically contained in four packets for IKEv1), securing the channel such that all following transactions are encrypted (see Phase 1 parameters on page 46).

The encrypted transactions contain the IKE authentication, since remote peers have yet to be authenticated. This stage of IKE authentication in IKEv2 can loosely be called Phase 1.5.

Phase 1.5

As part of this phase, IKE authentication must occur. IKE authentication consists of the following:

- The authentication payloads and Internet Security Association and Key Management Protocol (ISAKMP) identifier.
- The authentication method (RSA, PSK, ECDSA, or EAP).
- The IPsec SA parameters.

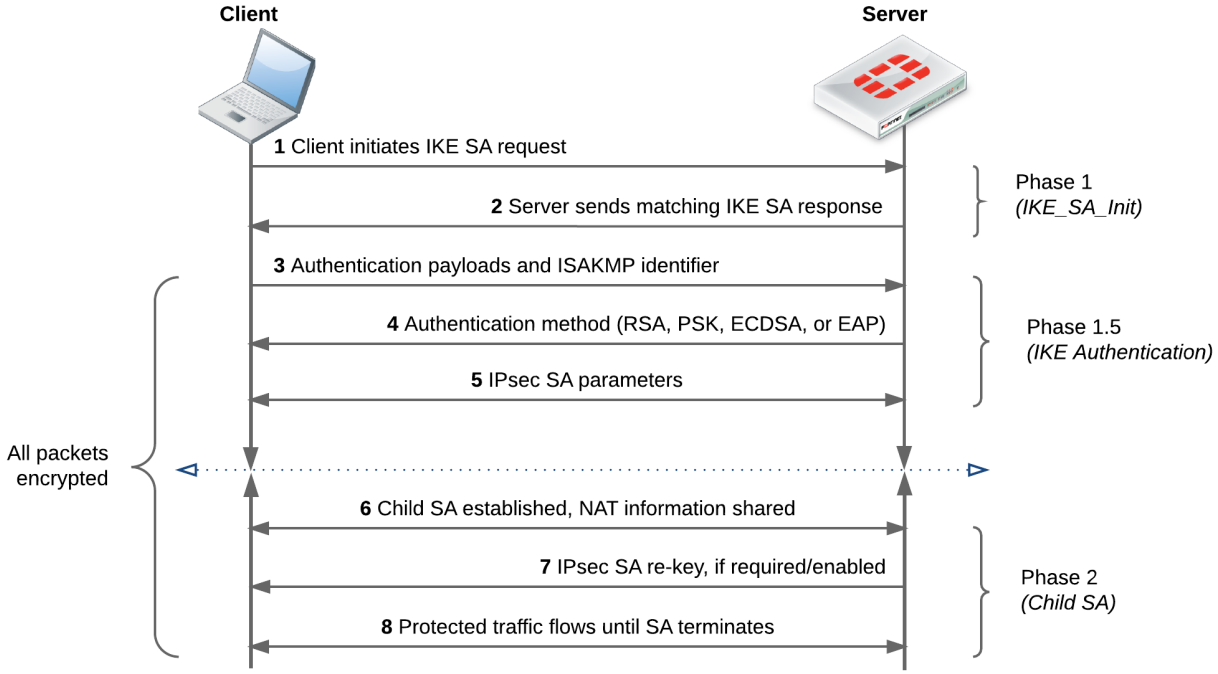
Due to the number of authentication methods potentially used, and SAs established, the overall IKEv2 negotiation can range from 4 packets (no EAP exchange at all) to many more.

At this point, both peers have a security association complete and ready to encrypt traffic.

Phase 2

In IKEv1, Phase 2 uses Quick mode to negotiate an IPsec SA between peers. In IKEv2, since the IPsec SA is already established, Phase 2 is essentially only used to negotiate “child” SAs, or to re-key an IPsec SA. That said, there are only two packets for each exchange of this type, similar to the exchange at the outset of Phase 1.5.

The entire IKEv2 process is demonstrated in the following diagram:



Support for IKEv2 session resumption

If a gateway loses connectivity to the network, clients can attempt to re-establish the lost session by presenting the ticket to the gateway (as described in RFC 5723). As a result, sessions can be resumed much faster, as DH exchange that is necessary to establish a brand new connection is skipped. This feature implements "ticket-by-value", whereby all information necessary to restore the state of a particular IKE SA is stored in the ticket and sent to the client.

IKEv2 asymmetric authentication

Asymmetric authentication allows both sides of an authentication exchange to use different authentication methods, for example the initiator may be using a shared key, while the responder may have a public signature key and certificate.

The command `authmethod-remote` is avilable under `config vpn ipsec phase1-interface`.

For more detailed information on authentication of the IKE SA, see RFC 5996 - Internet Key Exchange Protocol Version 2 (IKEv2).

IKEv2 Digital Signature Authentication support

FortiOS supports the use of Digital Signature authentication, which changes the format of the Authentication Data payload in order to support different signature methods.

Instead of just containing a raw signature value calculated as defined in the original IKE RFCs, the Auth Data now includes an ASN.1 formatted object that provides details on how the signature was calculated, such as the signature type, hash algorithm, and signature padding method.

For more detailed information on IKEv2 Digital Signature authentication, see RFC 7427 - Signature Authentication in the Internet Key Exchange Version 2 (IKEv2).

Unique IKE identifiers

When enabled, the following `phase1` CLI command (`enforce-unique-id`) requires all IPsec VPN clients to use a unique identifer when connecting.

CLI syntax

```
config vpn ipsec phase1
   edit <name>
      set enforce-unique-id {keep-new | keep-old | disable} Default is disable.
   next
end
```

Use `keep-new` to replace the old connnection if an ID collision is detected on the gateway.
Use `keep-old` to reject the new connection if an ID collision is detected.

IKEv2 ancillary RADIUS group authentication

This feature provides for the IDi information to be extracted from the IKEv2 AUTH exchange and sent to a RADIUS server, along with a fixed password (configurable via CLI only), to perform an additional group authentication step prior to tunnel establishment. The RADIUS server may return framed-IP-address, framed-ip-netmask, and dns-server attributes, which are then applied to the tunnel.

It should be noted, unlike Xauth or EAP, this feature does not perform individual user authentication, but rather treats all users on the gateway as a single group, and authenticates that group with RADIUS using a fixed password. This feature also works with RADIUS accounting, including the `phase1 acct-verify` option.

Syntax

```
config vpn ipsec phase1-interface
   edit <name>
      set mode-cfg enable
      set type dynamic
      set ike-version 2
      set group-authentication {enable | disable}
      set group-authentication-secret <password>
   next
end
```

IPsec VPN overview

This section provides a brief overview of IPsec technology and includes general information about how to configure IPsec VPNs using this guide.

The following topics are included in this section:

VPN configurations interact with the firewall component of the FortiGate unit. There must be a security policy in place to permit traffic to pass between the private network and the VPN tunnel.

Security policies for VPNs specify:

- The FortiGate interface that provides the physical connection to the remote VPN gateway, usually an interface connected to the Internet
- The FortiGate interface that connects to the private network
- IP addresses associated with data that has to be encrypted and decrypted
- Optionally, a schedule that restricts when the VPN can operate
- Optionally, the services (types of data) that can be sent

When the first packet of data that meets all of the conditions of the security policy arrives at the FortiGate unit, a VPN tunnel may be initiated and the encryption or decryption of data is performed automatically afterward. For more information, see Defining VPN security policies on page 1.

Where possible, you should create route-based VPNs. Generally, route-based VPNs are more flexible and easier to configure than policy-based VPNs — by default they are treated as interfaces. However, these two VPN types have different requirements that limit where they can be used.

Types of VPNs

FortiGate unit VPNs can be policy-based or route-based. There is little difference between the two types. In both cases, you specify Phase 1 and Phase 2 settings. However there is a difference in implementation. A route-based VPN creates a virtual IPsec network interface that applies encryption or decryption as needed to any traffic that it carries. That is why route-based VPNs are also known as interface-based VPNs. A policy-based VPN is implemented through a special security policy that applies the encryption you specified in the Phase 1 and Phase 2 settings.

Route-based VPNs

For a route-based VPN, you create two security policies between the virtual IPsec interface and the interface that connects to the private network. In one policy, the virtual interface is the source. In the other policy, the virtual interface is the destination. This creates bidirectional policies that ensure traffic will flow in both directions over the VPN.

A route-based VPN is also known as an interface-based VPN.

Each route-based IPsec VPN tunnel requires a virtual IPsec interface. As such, the amount of possible route-based IPsec VPNs is limited by the **system.interface** table size. The **system.interface** table size for most devices is 8192.

For a complete list of table sizes for all devices, refer to the Maximum Values table.

Policy-based VPNs

For a policy-based VPN, one security policy enables communication in both directions. You enable inbound and outbound traffic as needed within that policy, or create multiple policies of this type to handle different types of traffic differently. For example HTTPS traffic may not require the same level of scanning as FTP traffic.

A policy-based VPN is also known as a tunnel-mode VPN.

Comparing policy-based or route-based VPNs

For both VPN types you create Phase 1 and Phase 2 configurations. Both types are handled in the stateful inspection security layer, assuming there is no IPS or AV. For more information on the three security layers, see the FortiOS Troubleshooting guide.

The main difference is in the security policy.

You create a policy-based VPN by defining an IPSEC security policy between two network interfaces and associating it with the VPN tunnel (Phase 1) configuration.

You create a route-based VPN by creating a virtual IPsec interface. You then define a regular ACCEPT security policy to permit traffic to flow between the virtual IPsec interface and another network interface. And lastly, configure a static route to allow traffic over the VPN.

Where possible, you should create route-based VPNs. Generally, route-based VPNs are more flexible and easier to configure than policy-based VPNs — by default they are treated as interfaces. However, these two VPN types have different requirements that limit where they can be used.

Comparison of policy-based and route-based VPNs

Features	Policy-based	Route-based
Both NAT and transparent modes available	Yes	NAT mode only
L2TP-over-IPsec supported	Yes	Yes
GRE-over-IPsec supported	No	Yes
security policy requirements	Requires a security policy with IPSEC action that specifies the VPN tunnel	Requires only a simple security policy with ACCEPT action
Number of policies per VPN	One policy controls connections in both directions	A separate policy is required for connections in each direction

Planning your VPN

It is a good idea to plan the VPN configuration ahead of time. This will save time later and help you configure your VPN correctly.

All VPN configurations are comprised of numerous required and optional parameters. Before you begin, you need to determine:

- Where the IP traffic originates and where it needs to be delivered
- Which hosts, servers, or networks to include in the VPN
- Which VPN devices to include in the configuration
- Through which interfaces the VPN devices communicate
- Through which interfaces do private networks access the VPN gateways

Once you have this information, you can select a VPN topology that suits the network environment.

Network topologies

The topology of your network will determine how remote peers and clients connect to the VPN and how VPN traffic is routed.

VPN network topologies and brief descriptions

Topology	Description
Gateway-to-gateway configurations	Standard one-to-one VPN between two FortiGate units. See Gateway-to-gateway configurations on page 1.
Hub-and-spoke configurations	One central FortiGate unit has multiple VPNs to other remote FortiGate units. See Hub-and-spoke configurations on page 1.
Dynamic DNS configuration	One end of the VPN tunnel has a changing IP address and the other end must go to a dynamic DNS server for the current IP address before establishing a tunnel. See Dynamic DNS configuration on page 1.
FortiClient dialup-client configurations	Typically remote FortiClient dialup-clients use dynamic IP addresses through NAT devices. The FortiGate unit acts as a dialup server allowing dialup VPN connections from multiple sources. See FortiClient dialup-client configurations on page 1.
FortiGate dialup-client configurations	Similar to FortiClient dialup-client configurations but with more gateway-to-gateway settings such as unique user authentication for multiple users on a single VPN tunnel. See FortiGate dialup-client configurations on page 1.
Internet-browsing configuration	Secure web browsing performed by dialup VPN clients, and/or hosts behind a remote VPN peer. See Internet-browsing configuration on page 1.

Topology	Description
Redundant VPN configurations	Options for supporting redundant and partially redundant IPsec VPNs, using route-based approaches. See Redundant VPN configurations on page 1.
Transparent mode VPNs	In transparent mode, the FortiGate acts as a bridge with all incoming traffic being broadcast back out on all other interfaces. Routing and NAT must be performed on external routers. See Transparent mode VPNs on page 1.
L2TP and IPsec (Microsoft VPN)	Configure VPN for Microsoft Windows dialup clients using the built in L2TP software. Users do not have to install any See L2TP and IPsec (Microsoft VPN) on page 1.

These sections contain high-level configuration guidelines with cross-references to detailed configuration procedures. If you need more detail to complete a step, select the cross-reference in the step to drill-down to more detail. Return to the original procedure to complete the procedure. For a general overview of how to configure a VPN, see Planning your VPN .

General preparation steps

A VPN configuration defines relationships between the VPN devices and the private hosts, servers, or networks making up the VPN. Configuring a VPN involves gathering and recording the following information. You will need this information to configure the VPN.

- **The private IP addresses of participating hosts, servers, and/or networks.** These IP addresses represent the source addresses of traffic that is permitted to pass through the VPN. A IP source address can be an individual IP address, an address range, or a subnet address.
- **The public IP addresses of the VPN end-point interfaces.** The VPN devices establish tunnels with each other through these interfaces.
- **The private IP addresses associated with the VPN-device interfaces to the private networks.** Computers on the private networks behind the VPN gateways will connect to their VPN gateways through these interfaces.

How to use this guide to configure an IPsec VPN

This guide uses a task-based approach to provide all of the procedures needed to create different types of VPN configurations. Follow the step-by-step configuration procedures in this guide to set up the VPN.

The following configuration procedures are common to all IPsec VPNs:

1. Define the Phase 1 parameters that the FortiGate unit needs to authenticate remote peers or clients and establish a secure a connection. See Phase 1 parameters on page 46.
2. Define the Phase 2 parameters that the FortiGate unit needs to create a VPN tunnel with a remote peer or dialup client. See Phase 2 parameters on page 66.
3. Specify the source and destination addresses of IP packets that are to be transported through the VPN tunnel. See Defining policy addresses on page 1.

4. Create an IPsec security policy to define the scope of permitted services between the IP source and destination addresses. See Defining VPN security policies on page 1.

These steps assume you configure the FortiGate unit to generate unique IPsec encryption and authentication keys automatically. In situations where a remote VPN peer or client requires a specific IPsec encryption and authentication key, you must configure the FortiGate unit to use manual keys instead of performing Steps 1 and 2.

IPsec VPN in the web-based manager

To configure an IPsec VPN, use the general procedure below. With these steps, your FortiGate unit will automatically generate unique IPsec encryption and authentication keys. If a remote VPN peer or client requires a specific IPsec encryption or authentication key, you must configure your FortiGate unit to use manual keys instead.

1. Define Phase 1 parameters to authenticate remote peers and clients for a secure connection. See IPsec VPN in the web-based manager on page 32.
2. Define Phase 2 parameters to create a VPN tunnel with a remote peer or dialup client. See IPsec VPN in the web-based manager on page 32.
3. Create a security policy to permit communication between your private network and the VPN. Policy-based VPNs have an action of IPSEC, where for interface-based VPNs the security policy action is ACCEPT. See Defining VPN security policies on page 1.

The FortiGate unit implements the Encapsulated Security Payload (ESP) protocol. Internet Key Exchange (IKE) is performed automatically based on pre-shared keys or X.509 digital certificates. Interface mode, supported in NAT mode only, creates a virtual interface for the local end of a VPN tunnel.

This chapter contains the following sections:

Phase 1 configuration

To begin defining the Phase 1 configuration, go to **VPN > IPsec Tunnels** and select **Create New**. Enter a unique descriptive name for the VPN tunnel and follow the instructions in the VPN Creation Wizard.

The Phase 1 configuration mainly defines the ends of the IPsec tunnel. The remote end is the remote gateway with which the FortiGate unit exchanges IPsec packets. The local end is the FortiGate interface that sends and receives IPsec packets.

If you want to control how the IKE negotiation is processed when there is no traffic, as well as the length of time the FortiGate unit waits for negotiations to occur, you can use the `negotiation-timeout` and `auto-negotiate` commands in the CLI.

For more information, refer to Phase 2 parameters on page 66 and Phase 2 parameters on page 66.

Name	Type a name for the Phase 1 definition. The maximum name length is 15 characters for an interface mode VPN, 35 characters for a policy-based VPN. If **Remote Gateway** is **Dialup User**, the maximum name length is further reduced depending on the number of dialup tunnels that can be established: by 2 for up to 9 tunnels, by 3 for up to 99 tunnels, 4 for up to 999 tunnels, and so on. For a tunnel mode VPN, the name normally reflects where the remote connection originates. For a route-based tunnel, the FortiGate unit also uses the name for the virtual IPsec interface that it creates automatically.
Remote Gateway	Select the category of the remote connection: **Static IP Address** — If the remote peer has a static IP address. **Dialup User** — If one or more FortiClient or FortiGate dialup clients with dynamic IP addresses will connect to the FortiGate unit. **Dynamic DNS** — If a remote peer that has a domain name and subscribes to a dynamic DNS service will connect to the FortiGate unit.
IP Address	If you selected **Static IP Address**, enter the IP address of the remote peer.
Dynamic DNS	If you selected ***Dynamic DNS***, enter the domain name of the remote peer.
Local Interface	This option is available in NAT mode only. Select the name of the interface through which remote peers or dialup clients connect to the FortiGate unit. By default, the local VPN gateway IP address is the IP address of the interface that you selected.
Mode	**Main mode** — the Phase 1 parameters are exchanged in multiple rounds with encrypted authentication information. **Aggressive mode** — the Phase 1 parameters are exchanged in single message with authentication information that is not encrypted. When the remote VPN peer has a dynamic IP address and is authenticated by a pre-shared key, you must select Aggressive mode if there is more than one dialup phase1 configuration for the interface IP address. When the remote VPN peer has a dynamic IP address and is authenticated by a certificate, you must select Aggressive mode if there is more than one Phase 1 configuration for the interface IP address and these Phase 1 configurations use different proposals.
Authentication Method	Select **Preshared Key** or **RSA Signature**.

Pre-shared Key	If you selected **Pre-shared Key**, enter the pre-shared key that the FortiGate unit will use to authenticate itself to the remote peer or dialup client during Phase 1 negotiations. You must define the same key at the remote peer or client. The key must contain at least 6 printable characters. For optimum protection against currently known attacks, the key must consist of a minimum of 16 randomly chosen alphanumeric characters. The limit is 128 characters.
Certificate Name	If you selected **RSA Signature**, select the name of the server certificate that the FortiGate unit will use to authenticate itself to the remote peer or dialup client during Phase 1 negotiations. For information about obtaining and loading the required server certificate, see the FortiOS User Authentication guide.
Peer Options	Peer options are available to authenticate VPN peers or clients, depending on the **Remote Gateway** and **Authentication Method** settings.
Any peer ID	Accept the local ID of any remote VPN peer or client. The FortiGate unit does not check identifiers (local IDs). You can set **Mode** to **Aggressive** or **Main**. You can use this option with RSA Signature authentication. But, for highest security, configure a PKI user/group for the peer and set **Peer Options** to **Accept this peer certificate only**.
This peer ID	This option is available when **Aggressive Mode** is enabled. Enter the identifier that is used to authenticate the remote peer. This identifier must match the Local ID that the remote peer's administrator has configured. If the remote peer is a FortiGate unit, the identifier is specified in the **Local ID** field of the Advanced Phase 1 configuration. If the remote peer is a FortiClient user, the identifier is specified in the **Local ID** field, accessed by selecting **Config** in the **Policy** section of the VPN connection's **Advanced Settings**. In circumstances where multiple remote dialup VPN tunnels exist, each tunnel must have a peer ID set.

Peer ID from dialup group	Authenticate multiple FortiGate or FortiClient dialup clients that use unique identifiers and unique pre-shared keys (or unique pre-shared keys only) through the same VPN tunnel. You must create a dialup user group for authentication purposes. Select the group from the list next to the **Peer ID from dialup group** option. You must set **Mode** to **Aggressive** when the dialup clients use unique identifiers and unique pre-shared keys. If the dialup clients use unique pre-shared keys only, you can set **Mode** to **Main** if there is only one dialup Phase 1 configuration for this interface IP address.

Phase 1 advanced configuration settings

You can use the following advanced parameters to select the encryption and authentication algorithms that the FortiGate unit uses to generate keys for the IKE exchange. You can also use the following advanced parameters to ensure the smooth operation of Phase 1 negotiations.

These settings are mainly configured in the CLI, although some options are available after the tunnel is created using the VPN Creation Wizard (using the **Convert to Custom Tunnel** option).

If the FortiGate unit will act as a VPN client, and you are using security certificates for authentication, set the **Local ID** to the distinguished name (DN) of the local server certificate that the FortiGate unit will use for authentication purposes.

Note that, since FortiOS 5.4, an exact match is required to optimize IKE's gateway search utilizing binary trees. However, it is also possible to have partial matching of 'user.peer:cn' to match peers to gateways by performing a secondary match. When IKE receives IDi of type ASN1.DN, the first search is done with the whole DN string. If none is found, IKE will extract just the CN attribute value and perform a second search.

VXLAN over IPsec	Packets with VXLAN header are encapsulated within IPsec tunnel mode.
	
	To configure VXLAN over IPsec - CLI:
	
	`config vpn ipsec phase1-interface/phase1`
	`  edit ipsec`
	`  set interface <name>`
	`  set encapsulation vxlan/gre`
	`  set encapsulation-address ike/ipv4/ipv6`
	`  set encap-local-gw4 xxx.xxx.xxx.xxx`
	`  set encap-remote-gw xxx.xxx.xxx.xxx`
	`  next`
	`end`

IPsec tunnel idle timer	You can define an idle timer for IPsec tunnels. When no traffic has passed through the tunnel for the configured idle-timeout value, the IPsec tunnel will be flushed.
	To configure IPsec tunnel idle timeout - CLI:
	`config vpn ipsec phase1-interface`
	`edit p1`
	`set idle-timeout [enable | disable]`
	`set idle-timeoutinterval <integer>` //IPsec tunnel idle timeout in minutes (10 - 43200).
	`end`
	`end`

IPv6 Version	Select if you want to use IPv6 addresses for the remote gateway and interface IP addresses.
Local Gateway IP	Specify an IP address for the local end of the VPN tunnel. Select one of the following: **Main Interface IP** — The FortiGate unit obtains the IP address of the interface from the network interface settings. **Specify** — Enter a secondary address of the interface selected in the Phase 1 **Local Interface** field. You cannot configure Interface mode in a transparent mode VDOM.
Phase 1 Proposal	Select the encryption and authentication algorithms used to generate keys for protecting negotiations and add encryption and authentication algorithms as required. You need to select a minimum of one and a maximum of three combinations. The remote peer or client must be configured to use at least one of the proposals that you define. Select one of the following symmetric-key encryption algorithms: **DES** — Digital Encryption Standard, a 64-bit block algorithm that uses a 56-bit key. **3DES** — Triple-DES; plain text is encrypted three times by three keys. **AES128** — A 128-bit block algorithm that uses a 128-bit key. **AES192** — A 128-bit block algorithm that uses a 192-bit key. **AES256** — A 128-bit block algorithm that uses a 256-bit key. **ChaCha20/Poly1305**— A 128-bit block algorithm that uses a 128-bit key and a symmetric cipher. Only available for IKEv2.

	You can select either of the following message digests to check the authenticity of messages during an encrypted session: **MD5** — Message Digest 5. **SHA1** — Secure Hash Algorithm 1 - a 160-bit message digest. To specify one combination only, set the **Encryption** and **Authentication** options of the second combination to NULL. To specify a third combination, use the **Add** button beside the fields for the second combination.
Diffie-Hellman Group	Select one or more Diffie-Hellman groups from DH groups 1, 2, 5, and 14 through 21. At least one of the **Diffie-Hellman Group** settings on the remote peer or client must match one the selections on the FortiGate unit. Failure to match one or more DH groups will result in failed negotiations.
Keylife	Enter the time (in seconds) that must pass before the IKE encryption key expires. When the key expires, a new key is generated without interrupting service. The keylife can be from 120 to 172 800 seconds.
Local ID	If the FortiGate unit will act as a VPN client and you are using peer IDs for authentication purposes, enter the identifier that the FortiGate unit will supply to the VPN server during the Phase 1 exchange. If the FortiGate unit will act as a VPN client, and you are using security certificates for authentication, select the distinguished name (DN) of the local server certificate that the FortiGate unit will use for authentication purposes. If the FortiGate unit is a dialup client and will not be sharing a tunnel with other dialup clients (that is, the tunnel will be dedicated to this Fortinet dialup client), set **Mode** to **Aggressive**. Note that this Local ID value must match the peer ID value given for the remote VPN peer's Peer Options.

XAuth	This option supports the authentication of dialup clients. It is available for IKE v1 only. **Disable** — Select if you do not use XAuth. **Enable as Client** — If the FortiGate unit is a dialup client, enter the user name and password that the FortiGate unit will need to authenticate itself to the remote XAuth server. **Enable as Server** — This is available only if **Remote Gateway** is set to **Dialup User**. Dialup clients authenticate as members of a dialup user group. You must first create a user group for the dialup clients that need access to the network behind the FortiGate unit. You must also configure the FortiGate unit to forward authentication requests to an external RADIUS or LDAP authentication server. Select a **Server Type** setting to determine the type of encryption method to use between the FortiGate unit, the XAuth client and the external authentication server, and then select the user group from the User Group list.
Username	Enter the user name that is used for authentication.
Password	Enter the password that is used for authentication.
NAT Traversal	Select the check box if a NAT device exists between the local FortiGate unit and the VPN peer or client. The local FortiGate unit and the VPN peer or client must have the same NAT traversal setting (both selected or both cleared) to connect reliably. Additionally, you can force IPsec to use NAT traversal. If NAT is set to **Forced**, the FortiGate will use a port value of zero when constructing the NAT discovery hash for the peer. This causes the peer to think it is behind a NAT device, and it will use UDP encapsulation for IPsec, even if no NAT is present. This approach maintains interoperability with any IPsec implementation that supports the NAT-T RFC.
Keepalive Frequency	If you enabled **NAT-traversal**, enter a keepalive frequency setting.
Dead Peer Detection	Select this check box to reestablish VPN tunnels on idle connections and clean up dead IKE peers if required. You can use this option to receive notification whenever a tunnel goes up or down, or to keep the tunnel connection open when no traffic is being generated inside the tunnel. For example, in scenarios where a dialup client or dynamic DNS peer connects from an IP address that changes periodically, traffic may be suspended while the IP address changes. With **Dead Peer Detection** selected, you can use the `config vpn ipsec phase1` (tunnel mode) or `config vpn ipsec phase1-interface` (interface mode) CLI command to optionally specify a retry count and a retry interval.

IKEv1 fragmentation

UDP fragmentation can cause issues in IPsec when either the ISP or perimeter firewall(s) cannot pass or fragment the oversized UDP packets that occur when using a very large public security key (PSK). The result is that IPsec tunnels do not come up. The solution is IKE fragmentation.

For most configurations, enabling IKE fragmentation allows connections to automatically establish when they otherwise might have failed due to intermediate nodes dropping IKE messages containing large certificates, which typically push the packet size over 1500 bytes.

FortiOS will fragment a packet on sending if, and only if, all the following are true:

- Phase 1 contains "`set fragmentation enable`".
- The packet is larger than the minimum MTU (576 for IPv4, 1280 for IPv6).
- The packet is being re-transmitted.

By default, IKE fragmentation is enabled, but upon upgrading, any existing phase1-interface may have have "`set fragmentation disable`" added in order to preserve the existing behaviour of not supporting fragmentation.

Enabling or disabling IKE fragmentation - CLI

```
config vpn ipsec phase1-interface
   edit 1
      set fragmentation [enable | disable]
   next
end
```

IKEv2 fragmentation

With IKEv2, because RFC 7383 requires each fragment to be individually encrypted and authenticated, we would have to keep a copy of the unencrypted payloads around for each outgoing packet, in case the original single packet was never answered and we wanted to retry with fragments. With the following implementation, if the IKE payloads are greater than a configured threshold, the IKE packets are preemptively fragmented and encrypted.

CLI syntax

```
config vpn ipsec phase1-interface
   edit ike
      set ike-version 2
      set fragmentation [enable|disable]
      set fragmentation-mtu [500-16000]
   next
end
```

Phase 2 configuration

After IPsec Phase 1 negotiations end successfully, you begin Phase 2. You can configure the Phase 2 parameters to define the algorithms that the FortiGate unit may use to encrypt and transfer data for the remainder of the session. During Phase 2, you select specific IPsec security associations needed to implement security services and establish a tunnel.

The basic Phase 2 settings associate IPsec Phase 2 parameters with the Phase 1 configuration that specifies the remote end point of the VPN tunnel. In most cases, you need to configure only basic Phase 2 settings.

These settings are mainly configured in the CLI, although some options are available after the tunnel is created using the VPN Creation Wizard (using the **Convert to Custom Tunnel** option).

Name	Type a name to identify the Phase 2 configuration.
Phase 1	Select the Phase 1 tunnel configuration. For more information on configuring Phase 1, see Phase 1 configuration on page 32. The Phase 1 configuration describes how remote VPN peers or clients will be authenticated on this tunnel, and how the connection to the remote peer or client will be secured.
Advanced	Define advanced Phase 2 parameters. For more information, see Phase 2 advanced configuration settings below.

Phase 2 advanced configuration settings

In Phase 2, the FortiGate unit and the VPN peer or client exchange keys again to establish a secure communication channel between them. You select the encryption and authentication algorithms needed to generate keys for protecting the implementation details of Security Associations (SAs). These are called Phase 2 Proposal parameters. The keys are generated automatically using a Diffie-Hellman algorithm.

You can use a number of additional advanced Phase 2 settings to enhance the operation of the tunnel.

Phase 2 Proposal	Select the encryption and authentication algorithms that will be proposed to the remote VPN peer. You can specify up to three proposals. To establish a VPN connection, at least one of the proposals that you specify must match configuration on the remote peer. Initially there are two proposals. **Add** and **Delete** icons are next to the second **Authentication** field. It is invalid to set both **Encryption** and **Authentication** to **NULL**.
Encryption	Select a symmetric-key algorithms: **NULL** — Do not use an encryption algorithm. **DES** — Digital Encryption Standard, a 64-bit block algorithm that uses a 56-bit key. **3DES** — Triple-DES; plain text is encrypted three times by three keys. **AES128** — A 128-bit block algorithm that uses a 128-bit key. **AES192** — A 128-bit block algorithm that uses a 192-bit key. **AES256** — A 128-bit block algorithm that uses a 256-bit key. **ChaCha20/Poly1305**— A 128-bit block algorithm that uses a 128-bit key and a symmetric cipher. Only available for IKEv2.

Authentication	You can select either of the following message digests to check the authenticity of messages during an encrypted session: **NULL** — Do not use a message digest. **MD5** — Message Digest 5. **SHA1** — Secure Hash Algorithm 1 - a 160-bit message digest. To specify one combination only, set the **Encryption** and **Authentication** options of the second combination to NULL. To specify a third combination, use the **Add** button beside the fields for the second combination.
Enable replay detection	Replay attacks occur when an unauthorized party intercepts a series of IPsec packets and replays them back into the tunnel.
Enable perfect forward secrecy (PFS)	Perfect forward secrecy (PFS) improves security by forcing a new Diffie-Hellman exchange whenever keylife expires.
Diffie-Hellman Group	Select one Diffie-Hellman group (1, 2, 5, or 14 through 21). This must match the DH Group that the remote peer or dialup client uses.
Keylife	Select the method for determining when the Phase 2 key expires: **Seconds**, **KBytes**, or **Both**. If you select **Both**, the key expires when either the time has passed or the number of KB have been processed.
Autokey Keep Alive	Select the check box if you want the tunnel to remain active when no data is being processed.
Auto-negotiate	Enable the option if you want the tunnel to be automatically renegotiated when the tunnel expires.
DHCP-IPsec	Provide IP addresses dynamically to VPN clients. This is available for Phase 2 configurations associated with a dialup Phase 1 configuration. You also need configure a DHCP server or relay on the private network interface. You must configure the DHCP parameters separately. If you configure the DHCP server to assign IP addresses based on RADIUS user group attributes, you must also set the Phase 1 **Peer Options** to **Peer ID from dialup group** and select the appropriate user group. See Phase 1 configuration on page 32. If the FortiGate unit acts as a dialup server and you manually assigned FortiClient dialup clients VIP addresses that match the network behind the dialup server, selecting the check box will cause the FortiGate unit to act as a proxy for the dialup clients.

Quick Mode Selector	Specify the source and destination IP addresses to be used as selectors for IKE negotiations. If the FortiGate unit is a dialup server, keep the default value of 0.0.0.0/0 unless you need to circumvent problems caused by ambiguous IP addresses between one or more of the private networks making up the VPN. You can specify a single host IP address, an IP address range, or a network address. You may optionally specify source and destination port numbers and a protocol number. If you are editing an existing Phase 2 configuration, the **Source address** and **Destination address** fields are unavailable if the tunnel has been configured to use firewall addresses as selectors. This option exists only in the CLI.
Source address	If the FortiGate unit is a dialup server, enter the source IP address that corresponds to the local senders or network behind the local VPN peer (for example, `172.16.5.0/24` or `172.16.5.0/255.255.255.0` for a subnet, or `172.16.5.1/32` or `172.16.5.1/255.255.255.255` for a server or host, or `192.168.10.[80-100]` or `192.168.10.80-192.168.10.100` for an address range). A value of `0.0.0.0/0` means all IP addresses behind the local VPN peer. If the FortiGate unit is a dialup client, source address must refer to the private network behind the Fortinet dialup client.
Source port	Enter the port number that the local VPN peer uses to transport traffic related to the specified service (protocol number). The range is from 0 to 65535. To specify all ports, type `0`.
Destination address	Enter the destination IP address that corresponds to the recipients or network behind the remote VPN peer (for example, `192.168.20.0/24` for a subnet, or `172.16.5.1/32` for a server or host, or `192.168.10.[80-100]` for an address range). A value of `0.0.0.0/0` means all IP addresses behind the remote VPN peer.
Destination port	Enter the port number that the remote VPN peer uses to transport traffic related to the specified service (protocol number). To specify all ports, enter `0`.
Protocol	Enter the IP protocol number of the service. To specify all services, enter `0`.

FortiClient VPN

Use the **FortiClient VPN for OS X, Windows, and Android** VPN Wizard option when configuring an IPsec VPN for remote users to connect to the VPN tunnel using FortiClient.

When configuring a FortiClient VPN connection, the settings for Phase 1 and Phase 2 settings are automatically configured by the FortiGate unit. They are set to:

- Remote Gateway — Dialup User
- Mode — Aggressive
- Default settings for Phase 1 and 2 Proposals

- XAUTH Enable as Server (Auto)
- IKE mode-config will be enabled
- Peer Option — “Any peer ID”

The remainder of the settings use the current FortiGate defaults. Note that FortiClient settings need to match these FortiGate defaults. If you need to configure advanced settings for the FortiClient VPN, you must do so using the CLI.

Name	Enter a name for the FortiClient VPN.
Local Outgoing Interface	Select the local outgoing interface for the VPN.
Authentication Method	Select the type of authentication used when logging in to the VPN.
Preshared Key	If **Pre-shared Key** was selected in **Authentication Method**, enter the pre-shared key in the field provided.
User Group	Select a user group. You can also create a user group from the drop-down list by selecting **Create New**.
Address Range Start IP	Enter the start IP address for the DHCP address range for the client.
Address Range End IP	Enter the end IP address for the address range.
Subnet Mask	Enter the subnet mask.
Enable IPv4 Split Tunnel	Enabled by default, this option enables the FortiClient user to use the VPN to access internal resources while other Internet access is not sent over the VPN, alleviating potential traffic bottlenecks in the VPN connection. Disable this option to have all traffic sent through the VPN tunnel.
Accessible Networks	Select from a list of internal networks that the FortiClient user can access.
Client Options	These options affect how the FortiClient application behaves when connected to the FortiGate VPN tunnel. When enabled, a check box for the corresponding option appears on the VPN login screen in FortiClient, and is not enabled by default. **Save Password** - When enabled, if the user selects this option, their password is stored on the user’s computer and will automatically populate each time they connect to the VPN. **Auto Connect** - When enabled, if the user selects this option, when the FortiClient application is launched, for example after a reboot or system startup, FortiClient will automatically attempt to connect to the VPN tunnel. **Always Up (Keep Alive)** - When enabled, if the user selects this option, the FortiClient connection will not shut down. When not selected, during periods of inactivity, FortiClient will attempt to stay connected every three minutes for a maximum of 10 minutes.

Endpoint Registration	When selected, the FortiGate unit requests a registration key from FortiClient before a connection can be established. A registration key is defined by going to **System > Advanced**. For more information on FortiClient VPN connections to a FortiGate unit, see the **FortiClient Administration Guide**.
DNS Server	Select which DNS server to use for this VPN: **Use System DNS** — Use the same DNS servers as the FortiGate unit. These are configured at **Network > DNS**. This is the default option. **Specify** — Specify the IP address of a different DNS server.

Concentrator

In a hub-and-spoke configuration, policy-based VPN connections to a number of remote peers radiate from a single, central FortiGate unit. Site-to-site connections between the remote peers do not exist; however, you can establish VPN tunnels between any two of the remote peers through the FortiGate unit's "hub".

In a hub-and-spoke network, all VPN tunnels terminate at the hub. The peers that connect to the hub are known as "spokes". The hub functions as a concentrator on the network, managing all VPN connections between the spokes. VPN traffic passes from one tunnel to the other through the hub.

You define a concentrator to include spokes in the hub-and-spoke configuration. You create the concentrator in **VPN > IPsec Concentrator** and select **Create New**. A concentrator configuration specifies which spokes to include in an IPsec hub-and-spoke configuration.

Concentrator Name	Type a name for the concentrator.
Available Tunnels	A list of defined IPsec VPN tunnels. Select a tunnel from the list and then select the right arrow.
Members	A list of tunnels that are members of the concentrator. To remove a tunnel from the concentrator, select the tunnel and select the left arrow.

IPsec Monitor

You can use the IPsec Monitor to view activity on IPsec VPN tunnels and start or stop those tunnels. The display provides a list of addresses, proxy IDs, and timeout information for all active tunnels, including tunnel mode and route-based (interface mode) tunnels.

To view the IPsec monitor, go to **Monitor > IPsec Monitor**.

Tunnels are considered as "up" if at least one phase 2 selector is active. To avoid confusion, when a tunnel is down, **IPsec Monitor** will keep the **Phase 2 Selectors** column, but hide it by default and be replaced with **Phase 1** status column.

For dialup VPNs, the list provides status information about the VPN tunnels established by dialup clients, and their IP addresses.

For static IP or dynamic DNS VPNs, the list provides status and IP addressing information about VPN tunnels, active or not, to remote peers that have static IP addresses or domain names. You can also start and stop individual tunnels from the list.

Timeout field in IPsec Monitor page

The **Timeout** field in **Monitor > IPsec Monitor** shows the realtime timeout value for each VPN tunnel that is **Up** (tunnels that are **Down** show a timeout value of **0**).

Phase 1 parameters

This chapter provides detailed step-by-step procedures for configuring a FortiGate unit to accept a connection from a remote peer or dialup client. The Phase 1 parameters identify the remote peer or clients and supports authentication through preshared keys or digital certificates. You can increase access security further using peer identifiers, certificate distinguished names, group names, or the FortiGate extended authentication (XAuth) option for authentication purposes.

For more information on Phase 1 parameters in the web-based manager, see IPsec VPN in the web-based manager on page 32.

The information and procedures in this section do not apply to VPN peers that perform negotiations using manual keys.

The following topics are included in this section:

Overview

To configure IPsec Phase 1 settings, go to **VPN > IPsec Tunnels** and edit the **Phase 1 Proposal** (if it is not available, you may need to click the **Convert to Custom Tunnel** button).

IPsec Phase 1 settings define:

- The remote and local ends of the IPsec tunnel
- If Phase 1 parameters are exchanged in multiple rounds with encrypted authentication information (main mode) or in a single message with authentication information that is not encrypted (aggressive mode)
- If a preshared key or digital certificates will be used to authenticate the FortiGate unit to the VPN peer or dialup client
- If the VPN peer or dialup client is required to authenticate to the FortiGate unit. A remote peer or dialup client can authenticate by peer ID or, if the FortiGate unit authenticates by certificate, it can authenticate by peer certificate.
- The IKE negotiation proposals for encryption and authentication
- Optional XAuth authentication, which requires the remote user to enter a user name and password. A FortiGate VPN server can act as an XAuth server to authenticate dialup users. A FortiGate unit that is a dialup client can also be configured as an XAuth client to authenticate itself to the VPN server.

For all the Phase 1 web-based manager fields, see IPsec VPN in the web-based manager on page 32.

If you want to control how IKE is negotiated when there is no traffic, as well as the length of time the unit waits for negotiations to occur, use the `negotiation-timeout` and `auto-negotiate` commands in the CLI.

Defining the tunnel ends

To begin defining the Phase 1 configuration, go to **VPN > IPsec Tunnels** and select **Create New**. Enter a unique descriptive name for the VPN tunnel and follow the instructions in the VPN Creation Wizard.

The Phase 1 configuration mainly defines the ends of the IPsec tunnel. The remote end is the remote gateway with which the FortiGate unit exchanges IPsec packets. The local end is the FortiGate interface that sends and receives IPsec packets.

The remote gateway can be:

- A static IP address
- A domain name with a dynamic IP address
- A dialup client

A statically addressed remote gateway is the simplest to configure. You specify the IP address. Unless restricted in the security policy, either the remote peer or a peer on the network behind the FortiGate unit can bring up the tunnel.

If the remote peer has a domain name and subscribes to a dynamic DNS service, you need to specify only the domain name. The FortiGate unit performs a DNS query to determine the appropriate IP address. Unless restricted in the security policy, either the remote peer or a peer on the network behind the FortiGate unit can bring up the tunnel.

If the remote peer is a dialup client, only the dialup client can bring up the tunnel. The IP address of the client is not known until it connects to the FortiGate unit. This configuration is a typical way to provide a VPN for client PCs running VPN client software such as the FortiClient Endpoint Security application.

The local end of the VPN tunnel, the Local Interface, is the FortiGate interface that sends and receives the IPsec packets. This is usually the public interface of the FortiGate unit that is connected to the Internet (typically the WAN1 port). Packets from this interface pass to the private network through a security policy.

By default, the local VPN gateway is the IP address of the selected Local Interface. If you are configuring an interface mode VPN, you can optionally use a secondary IP address of the Local Interface as the local gateway.

Choosing Main mode or Aggressive mode

The FortiGate unit and the remote peer or dialup client exchange Phase 1 parameters in either Main mode or Aggressive mode. This choice does not apply if you use IKE version 2, which is available only for route-based configurations.

- In **Main** mode, the Phase 1 parameters are exchanged in multiple rounds with encrypted authentication information
- In **Aggressive** mode, the Phase 1 parameters are exchanged in a single message with unencrypted authentication information.

Although Main mode is more secure, you must select Aggressive mode if there is more than one dialup Phase 1 configuration for the interface IP address, and the remote VPN peer or client is authenticated using an identifier local ID. Aggressive mode might not be as secure as Main mode, but the advantage to Aggressive mode is that it

is faster than Main mode (since fewer packets are exchanged). Aggressive mode is typically used for remote access VPNs. But you would also use aggressive mode if one or both peers have dynamic external IP addresses. Descriptions of the peer options in this guide indicate whether Main or Aggressive mode is required.

Choosing the IKE version

If you create a route-based VPN, you have the option of selecting IKE version 2. Otherwise, IKE version 1 is used.

IKEv2, defined in RFC 4306, simplifies the negotiation process that creates the security association (SA).

If you select IKEv2:

- There is no choice in Phase 1 of Aggressive or Main mode.
- Extended Authentication (XAUTH) is not available.
- You can select only one Diffie-Hellman Group.
- You can utilize EAP and MOBIKE.

Repeated authentication in IKEv2

This feature provides the option to control whether a device requires its peer to re-authenticate or whether re-key is sufficient. It does not influence the re-authentication or re-key behavior of the device itself, which is controlled by the peer (with the default being to re-key). This solution is in response to RFC 4478. As described by the IETF, "the purpose of this is to limit the time that security associations (SAs) can be used by a third party who has gained control of the IPsec peer".

Syntax

```
config vpn ipsec phase1-interface
   edit p1
      set reauth [enable | disable]
   next
end
```

IKEv2 cookie notification for IKE_SA_INIT

IKEv2 offers an optional exchange within IKE_SA_INIT (the initial exchange between peers when establishing a secure tunnel) as a result of an inherent vulnerability in IPsec implementations, as described in RFC 5996.

Two expected attacks against IKE are state and CPU exhaustion, where the target is flooded with session initiation requests from forged IP addresses. These attacks can be made less effective if a responder uses minimal CPU and commits no state to an SA until it knows the initiator can receive packets at the address from which it claims to be sending them.

If the IKE_SA_INIT response includes the cookie notification, the initiator MUST then retry the IKE_SA_INIT request, and include the cookie notification containing the received data as the first payload, and all other payloads unchanged.

Upon detecting that the number of half-open IKEv2 SAs is above the threshold value, the VPN dialup server requires all future SA_INIT requests to include a valid cookie notification payload that the server sends back, in order to preserve CPU and memory resources.

For most devices, the threshold value is set to 500, half of the maximum 1,000 connections.

This feature is enabled by default in FortiOS 5.4.

IKEv2 Quick Crash Detection

There is support for IKEv2 Quick Crash Detection (QCD) as described in RFC 6290.

RFC 6290 describes a method in which an IKE peer can quickly detect that the gateway peer that it has and established an IKE session with has rebooted, crashed, or otherwise lost IKE state. When the gateway receives IKE messages or ESP packets with unknown IKE or IPsec SPIs, the IKEv2 protocol allows the gateway to send the peer an unprotected IKE message containing INVALID_IKE_SPI or INVALID_SPI notification payloads.

RFC 6290 introduces the concept of a QCD token, which is generated from the IKE SPIs and a private QCD secret, and exchanged between peers during the protected IKE AUTH exchange.

Adding Quick Crash Detection - CLI Syntax

```
config system settings
   set ike-quick-crash-detect [enable | disable]
end
```

IKEv1 Quick Crash Detection

Based on the IKEv2 QCD feature described above, IKEv1 QCD is implemented using a new IKE vendor ID, "Fortinet Quick Crash Detection", and so both endpoints must be FortiGate devices. The QCD token is sent in the Phase 1 exchange and must be encrypted, so this is only implemented for IKEv1 in Main mode (Aggressive mode is not supported as there is no available AUTH message in which to include the token).

Otherwise, the feature works the same as in IKEv2 (RFC 6290).

Authenticating the FortiGate unit

The FortiGate unit can authenticate itself to remote peers or dialup clients using either a pre-shared key or an RSA Signature (certificate).

Authenticating the FortiGate unit with digital certificates

To authenticate the FortiGate unit using digital certificates, you must have the required certificates installed on the remote peer and on the FortiGate unit. The signed server certificate on one peer is validated by the presence of the root certificate installed on the other peer. If you use certificates to authenticate the FortiGate unit, you can also require the remote peers or dialup clients to authenticate using certificates.

For more information about obtaining and installing certificates, see the FortiOS User Authentication guide.

Authenticating the FortiGate unit using digital certificates

1. Go to **VPN > IPsec Tunnels** and create the new custom tunnel or edit an existing tunnel.
2. Edit the **Phase 1 Proposal** (if it is not available, you may need to click the **Convert to Custom Tunnel** button):

Name	Enter a name that reflects the origination of the remote connection. For interface mode, the name can be up to 15 characters long.
Remote Gateway	Select the nature of the remote connection. Each option changes the available fields you must configure. For more information, see Authenticating the FortiGate unit on page 49.
Local Interface	Select the interface that is the local end of the IPsec tunnel. For more information, see Authenticating the FortiGate unit on page 49. The local interface is typically the WAN1 port.
Mode	Select a mode. It is easier to use Aggressive mode. In **Main** mode, parameters are exchanged in multiple encrypted rounds. In **Aggressive** mode, parameters are exchanged in a single unencrypted message. Aggressive mode must be used when the remote VPN peer or client has a dynamic IP address, or the remote VPN peer or client will be authenticated using an identifier (local ID). For more information, see Authenticating the FortiGate unit on page 49.
Authentication Method	Select **Signature.**
Certificate Name	Select the name of the server certificate that the FortiGate unit will use to authenticate itself to the remote peer or dialup client during Phase 1 negotiations. You must obtain and load the required server certificate before this selection. See the FortiOS User Authentication guide. If you have not loaded any certificates, use the certificate named **Fortinet_Factory**.
Peer Options	Peer options define the authentication requirements for remote peers or dialup clients. They are not for your FortiGate unit itself. See Authenticating the FortiGate unit on page 49.
Advanced	You can use the default settings for most Phase 1 configurations. Changes are required only if your network requires them. These settings includes IKE version, DNS server, P1 proposal encryption and authentication settings, and XAuth settings. See Authenticating the FortiGate unit on page 49.

3. If you are configuring authentication parameters for a dialup user group, optionally define extended authentication (XAuth) parameters in the Advanced section. See Authenticating the FortiGate unit on page 49.
4. Select **OK**.

Authenticating the FortiGate unit with a pre-shared key

The simplest way to authenticate a FortiGate unit to its remote peers or dialup clients is by means of a pre-shared key. This is less secure than using certificates, especially if it is used alone, without requiring peer IDs or extended authentication (XAuth). Also, you need to have a secure way to distribute the pre-shared key to the peers.

If you use pre-shared key authentication alone, all remote peers and dialup clients must be configured with the same pre-shared key. Optionally, you can configure remote peers and dialup clients with unique pre-shared keys. On the FortiGate unit, these are configured in user accounts, not in the phase_1 settings. For more information, see Authenticating the FortiGate unit on page 49.

The pre-shared key must contain at least 6 printable characters and best practices dictate that it be known only to network administrators. For optimum protection against currently known attacks, the key must consist of a minimum of 16 randomly chosen alphanumeric characters.

If you authenticate the FortiGate unit using a pre-shared key, you can require remote peers or dialup clients to authenticate using peer IDs, but not client certificates.

Authenticating the FortiGate unit with a pre-shared key

1. Go to **VPN > IPsec Tunnels** and create the new custom tunnel or edit an existing tunnel.
2. Edit the **Phase 1 Proposal** (if it is not available, you may need to click the **Convert to Custom Tunnel** button):

Name	Enter a name that reflects the origination of the remote connection.
Remote Gateway	Select the nature of the remote connection. For more information, see Authenticating the FortiGate unit on page 49.
Local Interface	Select the interface that is the local end of the IPsec tunnel. For more information, see Authenticating the FortiGate unit on page 49. The local interface is typically the WAN1 port.
Mode	Select Main or Aggressive mode. In **Main** mode, the Phase 1 parameters are exchanged in multiple rounds with encrypted authentication information. In **Aggressive** mode, the Phase 1 parameters are exchanged in single message with authentication information that is not encrypted. When the remote VPN peer or client has a dynamic IP address, or the remote VPN peer or client will be authenticated using an identifier (local ID), you must select Aggressive mode if there is more than one dialup Phase 1 configuration for the interface IP address. For more information, see Authenticating the FortiGate unit on page 49.
Authentication Method	Select **Pre-shared Key**.

Pre-shared Key	Enter the preshared key that the FortiGate unit will use to authenticate itself to the remote peer or dialup client during Phase 1 negotiations. You must define the same value at the remote peer or client. The key must contain at least 6 printable characters and best practices dictate that it only be known by network administrators. For optimum protection against currently known attacks, the key must consist of a minimum of 16 randomly chosen alphanumeric characters.
Peer options	Peer options define the authentication requirements for remote peers or dialup clients, not for the FortiGate unit itself. You can require the use of peer IDs, but not client certificates. For more information, see Authenticating the FortiGate unit on page 49.
Advanced	You can retain the default settings unless changes are needed to meet your specific requirements. See Authenticating the FortiGate unit on page 49.

3. If you are configuring authentication parameters for a dialup user group, optionally define extended authentication (XAuth) parameters. See Authenticating the FortiGate unit on page 49.
4. Select **OK**.

Authenticating remote peers and clients

Certificates or pre-shared keys restrict who can access the VPN tunnel, but they do not identify or authenticate the remote peers or dialup clients. You have the following options for authentication:

Methods of authenticating remote VPN peers

Certificates or Pre-shared key	Local ID	User account pre-shared keys	Reference
Certificates			See Enabling VPN access for specific certificate holders on page 53.
Either	**X**		See Enabling VPN access by peer identifier on page 55.
Pre-shared key		**X**	See Enabling VPN access with user accounts and pre-shared keys on page 56.
Pre-shared key	**X**	**X**	See Enabling VPN access with user accounts and pre-shared keys on page 56.

Repeated authentication in Internet Key Exchange (IKEv2) protocol

This feature provides the option to control whether a device requires its peer to re-authenticate or whether re-key is sufficient. It does not influence the re-authentication or re-key behavior of the device itself, which is controlled by the peer (with the default being to re-key).

This solution is in response to RFC 4478. This solution is intended to limit the time that security associations (SAs) can be used by a third party who has gained control of the IPsec peer.

CLI syntax:

```
config vpn ipsec phase1-interface
   edit p1
      set reauth [enable | disable]
   next
end
```

disable: Disable IKE SA re-authentication.

enable: Enable IKE SA re-authentication.

Enabling VPN access for specific certificate holders

When a VPN peer or dialup client is configured to authenticate using digital certificates, it sends the Distinguished Name (DN) of its certificate to the FortiGate unit. This DN can be used to allow VPN access for the certificate holder. That is, a FortiGate unit can be configured to deny connections to all remote peers and dialup clients except the one having the specified DN.

Before you begin

The following procedures assume that you already have an existing Phase 1 configuration (see Authenticating remote peers and clients on page 52). Follow the procedures below to add certificate-based authentication parameters to the existing configuration.

Before you begin, you must obtain the certificate DN of the remote peer or dialup client. If you are using the FortiClient application as a dialup client, refer to FortiClient online help for information about how to view the certificate DN. To view the certificate DN of a FortiGate unit, see Viewing server certificate information and obtaining the local DN on page 54.

Use the `config user peer` CLI command to load the DN value into the FortiGate configuration. For example, if a remote VPN peer uses server certificates issued by your own organization, you would enter information similar to the following:

```
config user peer
   edit DN_FG1000
      set cn 192.168.2.160
      set cn-type ipv4
end
```

The value that you specify to identify the entry (for example, DN_FG1000) is displayed in the **Accept this peer certificate only** list in the IPsec Phase 1 configuration when you return to the web-based manager.

If the remote VPN peer has a CA-issued certificate to support a higher level of credibility, you would enter information similar to the following in the CLI:

```
config user peer
   edit CA_FG1000
      set ca CA_Cert_1
      set subject FG1000_at_site1
end
```

The value that you specify to identify the entry (for example, CA_FG1000) is displayed in the Accept this peer certificate only list in the IPsec Phase 1 configuration when you return to the web-based manager. For more information about these CLI commands, see the “user” chapter of the FortiGate CLI Reference.

A group of certificate holders can be created based on existing user accounts for dialup clients. To create the user accounts for dialup clients, see the “User” chapter of the FortiGate Administration Guide. To create the certificate group afterward, use the `config user peergrp` CLI command. See the “user” chapter of the FortiGate CLI Reference.

Viewing server certificate information and obtaining the local DN

1. Go to **System > Certificates**.
2. Note the CN value in the **Subject** field (for example, `CN = 172.16.10.125`, `CN = info@fortinet.com`, or `CN = www.example.com`).

Viewing CA root certificate information and obtaining the CA certificate name

1. Go to **System > Certificates > CA Certificates**.
2. Note the value in the **Name** column (for example, `CA_Cert_1`).

Configuring certificate authentication for a VPN

With peer certificates loaded, peer users and peer groups defined, you can configure your VPN to authenticate users by certificate.

Enabling access for a specific certificate holder or a group of certificate holders

1. At the FortiGate VPN server, go to **VPN > IPsec Tunnels** and create the new custom tunnel or edit an existing tunnel.
2. Edit the **Phase 1 Proposal** (if it is not available, you may need to click the **Convert to Custom Tunnel** button).
3. From the **Authentication Method** list, select **RSA Signature**.
4. From the **Certificate Name** list, select the name of the server certificate that the FortiGate unit will use to authenticate itself to the remote peer or dialup client
5. Under **Peer Options**, select one of these options:
   - To accept a specific certificate holder, select **Accept this peer certificate only** and select the name of the certificate that belongs to the remote peer or dialup client. The certificate DN must be added to the FortiGate configuration through CLI commands before it can be selected here. See Before you begin on page 53.
   - To accept dialup clients who are members of a certificate group, select **Accept this peer certificate group only** and select the name of the group. The group must be added to the FortiGate configuration through CLI commands before it can be selected here. See Before you begin on page 53.
6. If you want the FortiGate VPN server to supply the DN of a local server certificate for authentication purposes, select **Advanced** and then from the **Local ID** list, select the DN of the certificate that the FortiGate VPN server is to use.
7. Select **OK**.

Enabling VPN access by peer identifier

Whether you use certificates or pre-shared keys to authenticate the FortiGate unit, you can require that remote peers or clients have a particular peer ID. This adds another piece of information that is required to gain access to the VPN. More than one FortiGate/FortiClient dialup client may connect through the same VPN tunnel when the dialup clients share a preshared key and assume the same identifier.

In circumstances where multiple remote dialup VPN tunnels exist, each tunnel must have a peer ID set.

A peer ID, also called local ID, can be up to 63 characters long containing standard regular expression characters. Local ID is set in phase1 Aggressive Mode configuration.

You cannot require a peer ID for a remote peer or client that uses a pre-shared key and has a static IP address.

Authenticating remote peers or dialup clients using one peer ID

1. At the FortiGate VPN server, go to **VPN > IPsec Tunnels** and create the new custom tunnel or edit an existing tunnel.
2. Edit the **Phase 1 Proposal** (if it is not available, you may need to click the **Convert to Custom Tunnel** button).
3. Select **Aggressive** mode in any of the following cases:
   - The FortiGate VPN server authenticates a FortiGate dialup client that uses a dedicated tunnel
   - A FortiGate unit has a dynamic IP address and subscribes to a dynamic DNS service
   - FortiGate/FortiClient dialup clients sharing the same preshared key and local ID connect through the same VPN tunnel
4. For the **Peer Options**, select **This peer ID** and type the identifier into the corresponding field.
5. Select **OK**.

Assigning an identifier (local ID) to a FortiGate unit

Use this procedure to assign a peer ID to a FortiGate unit that acts as a remote peer or dialup client.

1. Go to **VPN > IPsec Tunnels** and create the new custom tunnel or edit an existing tunnel.
2. Edit the **Phase 1 Proposal** (if it is not available, you may need to click the **Convert to Custom Tunnel** button).
3. Select **Advanced**.
4. In the **Local ID** field, type the identifier that the FortiGate unit will use to identify itself.
5. Set **Mode** to **Aggressive** if any of the following conditions apply:
   - The FortiGate unit is a dialup client that will use a unique ID to connect to a FortiGate dialup server through a dedicated tunnel.
   - The FortiGate unit has a dynamic IP address, subscribes to a dynamic DNS service, and will use a unique ID to connect to the remote VPN peer through a dedicated tunnel.
   - The FortiGate unit is a dialup client that shares the specified ID with multiple dialup clients to connect to a FortiGate dialup server through the same tunnel.
6. Select **OK**.

Configuring the FortiClient application

Follow this procedure to add a peer ID to an existing FortiClient configuration:

1. Start the FortiClient application.
2. Go to **VPN > Connections**, select the existing configuration.
3. Select **Advanced > Edit > Advanced**.
4. Under **Policy**, select **Config**.
5. In the **Local ID** field, type the identifier that will be shared by all dialup clients. This value must match the **This peer ID** value that you specified previously in the Phase 1 gateway configuration on the FortiGate unit.
6. Select **OK** to close all dialog boxes.
7. Configure all dialup clients the same way using the same preshared key and local ID.

Enabling VPN access with user accounts and pre-shared keys

You can permit access only to remote peers or dialup clients that have pre-shared keys and/or peer IDs configured in user accounts on the FortiGate unit.

If you want two VPN peers (or a FortiGate unit and a dialup client) to accept reciprocal connections based on peer IDs, you must enable the exchange of their identifiers when you define the Phase 1 parameters.

The following procedures assume that you already have an existing Phase 1 configuration (see Authenticating remote peers and clients on page 52). Follow the procedures below to add ID checking to the existing configuration.

Before you begin, you must obtain the identifier (local ID) of the remote peer or dialup client. If you are using the FortiClient Endpoint Security application as a dialup client, refer to the Authenticating FortiClient Dialup Clients Technical Note to view or assign an identifier. To assign an identifier to a FortiGate dialup client or a FortiGate unit that has a dynamic IP address and subscribes to a dynamic DNS service, see Assigning an identifier (local ID) to a FortiGate unit on page 55.

If required, a dialup user group can be created from existing user accounts for dialup clients. To create the user accounts and user groups, see the User Authentication handbook chapter.

The following procedure supports FortiGate/FortiClient dialup clients that use unique preshared keys and/or peer IDs. The client must have an account on the FortiGate unit and be a member of the dialup user group.

The dialup user group must be added to the FortiGate configuration before it can be selected. For more information, see the User Authentication handbook chapter.

The FortiGate dialup server compares the local ID that you specify at each dialup client to the FortiGate user-account user name. The dialup-client preshared key is compared to a FortiGate user-account password.

Authenticating dialup clients using unique preshared keys and/or peer IDs

1. At the FortiGate VPN server, go to **VPN > IPsec Tunnels** and create the new custom tunnel or edit an existing tunnel.
2. Edit the **Phase 1 Proposal** (if it is not available, you may need to click the **Convert to Custom Tunnel** button).
3. If the clients have unique peer IDs, set **Mode** to **Aggressive**.
4. Clear the **Pre-shared Key** field.
   The user account password will be used as the preshared key.
5. Select **Peer ID from dialup group** and then select the group name from the list of user groups.
6. Select **OK**.

Follow this procedure to add a unique pre-shared key and unique peer ID to an existing FortiClient configuration.

Configuring FortiClient - pre-shared key and peer ID

1. Start the FortiClient Endpoint Security application.
2. Go to **VPN > Connections**, select the existing configuration.
3. Select **Advanced > Edit**.
4. In the **Preshared Key** field, type the FortiGate password that belongs to the dialup client (for example, `1234546`).
   The user account password will be used as the preshared key.
5. Select **Advanced**.
6. Under **Policy**, select **Config**.
7. In the **Local ID** field, type the FortiGate user name that you assigned previously to the dialup client (for example, `FortiClient1`).
8. Select **OK** to close all dialog boxes.

Configure all FortiClient dialup clients this way using unique preshared keys and local IDs.

Follow this procedure to add a unique pre-shared key to an existing FortiClient configuration.

Configuring FortiClient - preshared key only

1. Start the FortiClient Endpoint Security application.
2. Go to **VPN > Connections**, select the existing configuration
3. Select **Advanced > Edit**.
4. In the **Preshared Key** field, type the user name, followed by a “+” sign, followed by the password that you specified previously in the user account settings on the FortiGate unit (for example, `FC2+1FG6LK`)
5. Select **OK** to close all dialog boxes.

Configure all the FortiClient dialup clients this way using their unique peer ID and pre-shared key values.

Defining IKE negotiation parameters

In Phase 1, the two peers exchange keys to establish a secure communication channel between them. As part of the Phase 1 process, the two peers authenticate each other and negotiate a way to encrypt further communications for the duration of the session. The Phase 1 Proposal parameters select the encryption and authentication algorithms that are used to generate keys for protecting negotiations.

The IKE negotiation parameters determine:

- Which encryption algorithms may be applied for converting messages into a form that only the intended recipient can read
- Which authentication hash may be used for creating a keyed hash from a preshared or private key
- Which Diffie-Hellman group (DH Group) will be used to generate a secret session key

Phase 1 negotiations (in main mode or aggressive mode) begin as soon as a remote VPN peer or client attempts to establish a connection with the FortiGate unit. Initially, the remote peer or dialup client sends the FortiGate unit a list of potential cryptographic parameters along with a session ID. The FortiGate unit compares those parameters to its own list of advanced Phase 1 parameters and responds with its choice of matching parameters

to use for authenticating and encrypting packets. The two peers handle the exchange of encryption keys between them, and authenticate the exchange through a preshared key or a digital signature.

Generating keys to authenticate an exchange

The FortiGate unit supports the generation of secret session keys automatically using a Diffie-Hellman algorithm. These algorithms are defined in RFC 2409. The **Keylife** setting in the **Phase 1 Proposal** area determines the amount of time before the Phase 1 key expires. Phase 1 negotiations are re-keyed automatically when there is an active security association. See Dead Peer Detection on page 61.

You can enable or disable automatic re-keying between IKE peers through the `phase1-rekey` attribute of the `config system global` CLI command. For more information, see the "System" chapter of the FortiGate CLI Reference.

When in FIPS-CC mode, the FortiGate unit requires DH key exchange to use values at least 3072 bits long. However most browsers need the key size set to 1024. You can set the minimum size of the DH keys in the CLI.

```
config system global
   set dh-params 3072
end
```

When you use a preshared key (shared secret) to set up two-party authentication, the remote VPN peer or client and the FortiGate unit must both be configured with the same preshared key. Each party uses a session key derived from the Diffie-Hellman exchange to create an authentication key, which is used to sign a known combination of inputs using an authentication algorithm (such as HMAC-MD5, HMAC-SHA-1, or HMAC-SHA-256). Hash-based Message Authentication Code (HMAC) is a method for calculating an authentication code using a hash function plus a secret key, and is defined in RFC 2104. Each party signs a different combination of inputs and the other party verifies that the same result can be computed.

For information regarding NP accelerated offloading of IPsec VPN authentication algorithms, please refer to the Hardware Acceleration handbook chapter.

When you use preshared keys to authenticate VPN peers or clients, you must distribute matching information to all VPN peers and/or clients whenever the preshared key changes.

As an alternative, the remote peer or dialup client and FortiGate unit can exchange digital signatures to validate each other's identity with respect to their public keys. In this case, the required digital certificates must be installed on the remote peer and on the FortiGate unit. By exchanging certificate DNs, the signed server certificate on one peer is validated by the presence of the root certificate installed on the other peer.

The following procedure assumes that you already have a Phase 1 definition that describes how remote VPN peers and clients will be authenticated when they attempt to connect to a local FortiGate unit. For information about the Local ID and XAuth options, see Defining IKE negotiation parameters on page 57 and Defining IKE negotiation parameters on page 57. Follow this procedure to add IKE negotiation parameters to the existing definition.

Defining IKE negotiation parameters

1. Go to **VPN > IPsec Tunnels** and create the new custom tunnel or edit an existing tunnel.
2. Edit the **Phase 1 Proposal** (if it is not available, you may need to click the **Convert to Custom Tunnel** button).
3. Select **Phase 1 Proposal** and include the appropriate entries as follows:

Phase 1 Proposal	Select the encryption and authentication algorithms that will be used to generate keys for protecting negotiations. Add or delete encryption and authentication algorithms as required. Select a minimum of one and a maximum of three combinations. The remote peer must be configured to use at least one of the proposals that you define. It is invalid to set both **Encryption** and **Authentication** to null.
Encryption	Select a symmetric-key algorithms: **NULL** — Do not use an encryption algorithm. **DES** — Digital Encryption Standard, a 64-bit block algorithm that uses a 56-bit key. **3DES** — Triple-DES; plain text is encrypted three times by three keys. **AES128** — A 128-bit block algorithm that uses a 128-bit key. **AES192** — A 128-bit block algorithm that uses a 192-bit key. **AES256** — A 128-bit block algorithm that uses a 256-bit key. **ChaCha20/Poly1305**— A 128-bit block algorithm that uses a 128-bit key and a symmetric cipher. Only available for IKEv2.
Authentication	You can select either of the following message digests to check the authenticity of messages during an encrypted session: **NULL** — Do not use a message digest. **MD5** — Message Digest 5. **SHA1** — Secure Hash Algorithm 1 - a 160-bit message digest. To specify one combination only, set the **Encryption** and **Authentication** options of the second combination to NULL. To specify a third combination, use the **Add** button beside the fields for the second combination. For information regarding NP accelerated offloading of IPsec VPN authentication algorithms, please refer to the Hardware Acceleration handbook chapter.

Diffie-Hellman Group	Select one or more Diffie-Hellman groups from DH groups 1, 2, 5, 14 through 21, and 27 through 30. When using aggressive mode, DH groups cannot be negotiated. By default, DH group 14 is selected, to provide sufficient protection for stronger cipher suites that include AES and SHA2. If you select multiple DH groups, the order they appear in the configuration is the order in which they are negotiates. If both VPN peers (or a VPN server and its client) have static IP addresses and use aggressive mode, select a single DH group. The setting on the FortiGate unit must be identical to the setting on the remote peer or dialup client. When the remote VPN peer or client has a dynamic IP address and uses aggressive mode, select up to three DH groups on the FortiGate unit and one DH group on the remote peer or dialup client. The setting on the remote peer or dialup client must be identical to one of the selections on the FortiGate unit. If the VPN peer or client employs main mode, you can select multiple DH groups. At least one of the settings on the remote peer or dialup client must be identical to the selections on the FortiGate unit.
Keylife	Type the amount of time (in seconds) that will be allowed to pass before the IKE encryption key expires. When the key expires, a new key is generated without interrupting service. The keylife can be from 120 to 172800 seconds.
Nat-traversal	Enable this option if a NAT device exists between the local FortiGate unit and the VPN peer or client. The local FortiGate unit and the VPN peer or client must have the same NAT traversal setting (both selected or both cleared). When in doubt, enable NAT-traversal. See NAT traversal on page 60.
Keepalive Frequency	If you enabled NAT traversal, enter a keepalive frequency setting. The value represents an interval from 0 to 900 seconds where the connection will be maintained with no activity. For additional security this value must be as low as possible. See NAT keepalive frequency on page 61.
Dead Peer Detection	Enable this option to reestablish VPN tunnels on idle connections and clean up dead IKE peers if required. This feature minimizes the traffic required to check if a VPN peer is available or unavailable (dead). See Dead Peer Detection on page 61.

NAT traversal

Network Address Translation (NAT) is a way to convert private IP addresses to publicly routable Internet addresses and vise versa. When an IP packet passes through a NAT device, the source or destination address in the IP header is modified. FortiGate units support NAT version 1 (encapsulate on port 500 with non-IKE marker), version 3 (encapsulate on port 4500 with non-ESP marker), and compatible versions.

NAT cannot be performed on IPsec packets in ESP tunnel mode because the packets do not contain a port number. As a result, the packets cannot be demultiplexed. To work around this, the FortiGate unit provides a way to protect IPsec packet headers from NAT modifications. When the Nat-traversal option is enabled, outbound

encrypted packets are wrapped inside a UDP IP header that contains a port number. This extra encapsulation allows NAT devices to change the port number without modifying the IPsec packet directly.

To provide the extra layer of encapsulation on IPsec packets, the Nat-traversal option must be enabled whenever a NAT device exists between two FortiGate VPN peers or a FortiGate unit and a dialup client such as FortiClient. On the receiving end, the FortiGate unit or FortiClient removes the extra layer of encapsulation before decrypting the packet.

Additionally, you can force IPsec to use NAT traversal. If NAT is set to **Forced**, the FortiGate will use a port value of zero when constructing the NAT discovery hash for the peer. This causes the peer to think it is behind a NAT device, and it will use UDP encapsulation for IPsec, even if no NAT is present. This approach maintains interoperability with any IPsec implementation that supports the NAT-T RFC.

NAT keepalive frequency

When a NAT device performs network address translation on a flow of packets, the NAT device determines how long the new address will remain valid if the flow of traffic stops (for example, the connected VPN peer may be idle). The device may reclaim and reuse a NAT address when a connection remains idle for too long.

To work around this, when you enable NAT traversal specify how often the FortiGate unit sends periodic keepalive packets through the NAT device in order to ensure that the NAT address mapping does not change during the lifetime of a session. To be effective, the keepalive interval must be smaller than the session lifetime value used by the NAT device.

The keepalive packet is a 138-byte ISAKMP exchange.

Dead Peer Detection

Sometimes, due to routing issues or other difficulties, the communication link between a FortiGate unit and a VPN peer or client may go down. Packets could be lost if the connection is left to time out on its own. The FortiGate unit provides a mechanism called Dead Peer Detection (DPD), sometimes referred to as gateway detection or ping server, to prevent this situation and reestablish IKE negotiations automatically before a connection times out: the active Phase 1 security associations are caught and renegotiated (rekeyed) before the Phase 1 encryption key expires.

By default, Dead Peer Detection sends probe messages every five seconds by default (see `dpd-retryinterval` in the FortiGate CLI Reference). If you are experiencing high network traffic, you can experiment with increasing the ping interval. However longer intervals will require more traffic to detect dead peers which will result in more traffic.

In the web-based manager, the Dead Peer Detection option can be enabled when you define advanced Phase 1 options. The `config vpn ipsec phase1` CLI command supports additional options for specifying a retry count and a retry interval.

For more information about these commands and the related `config router gwdetect` CLI command, see the FortiGate CLI Reference.

For example, enter the following CLI commands to configure dead peer detection on the existing IPsec Phase 1 configuration called `test` to use 15 second intervals and to wait for 3 missed attempts before declaring the peer dead and taking action.

```
config vpn ipsec phase1-interface
   edit <value>
      set dpd [disable | on-idle | on-demand]
      set dpd-retryinveral 15
      set dpd-retrycount 3
```

```
    next
end
```

The default for `vpn ipsec phase1 dpd` is `on-idle` when the `type` is `dynamic` to encourage dialup server configurations to more pro-actively delete tunnels if the peer goes away.

DPD scalability

On a dial-up server, if a multitude of VPN connections are idle, the increased DPD exchange could negatively impact the performance/load of the daemon. For this reason, an option is available in the CLI to send DPD passively in a mode called "on-demand".

- When there is no traffic and the last DPD-ACK had been received, IKE will not send DPDs periodically.
- IKE will only send out DPDs if there are outgoing packets to send but no inbound packets had since been received.

Syntax

Set DPD to `on-demand` to trigger DPD when IPsec traffic is sent but no reply is received from the peer.

```
config vpn ipsec phase1-interface
   edit <value>
      set dpd [disable | on-idle | on-demand]
   next
end
```

Certificate key size control

Proxy will choose the same SSL key size as the HTTPS server. If the key size from the server is 512, the proxy will choose 1024. If the key size is bigger than 1024, the proxy will choose 2048.

As a result, the `firewall ssl-ssh-profile` commands `certname-rsa`, `certname-dsa`, and `certname-ecdsa` have been replaced with more specific key size control commands under `vpn certificate setting`.

CLI syntax

```
config vpn certificate setting
   set certname-rsa1024 <name>
   set certname-rsa2048 <name>
   set certname-dsa1024 <name>
   set certname-dsa2048 <name>
   set certname-ecdsa256 <name>
   set certname-ecdsa384 <name>
end
```

Quantum resistant IKEv2 SA negotiation

An IKEv2 extension is available that changes the key generation mechanism to include a Post-quantum Pre-shared Key.

The addition of PPK in the calculation means that even if a quantum computer can break the Diffie-Hellman calculation to derive the DH-generated secret key, the inclusion of the PPK in the key generation algorithm means that the attacker is still unable to derive the keys used to authenticate the IKE SA negotiation (and so cannot impersonate either party in the negotiation) nor the keys used in negotiating an IPsec SA (or IKE SA).

Syntax

```
config vpn ipsec phase1-interface
   edit <name>
      set ike-version 2
      set type dynamic
      set ppk {disable | allow | require}
      set ppk-secret <ASCII string or hexadecimal encoded with a leading 0x>
      set ppk-identity <string>
   next
end
config user local
   edit <name>
      set type password
      set ppk-secret <ASCII string or hexadecimal encoded with a leading 0x>
   next
end
```

For troubleshooting, `diagnose vpn ike ga list` can indicate whether PPK was negotiated.

- The 'PPK' at the gateway level indicates whether PPK was negotiated during the initial IKE SA negotiation.
- The 'PPK' at the IKE SA level indicates whether PPK was negotiated on this IKE SA.
- The 'child' at the IKE SA level indicates whether the IKE SA is an initial IKE SA or whether it is a child IKE SA. The above has 'child: no' and so it is initial IKE SA.

Using XAuth authentication

Extended authentication (XAuth) increases security by requiring the remote dialup client user to authenticate in a separate exchange at the end of Phase 1. XAuth draws on existing FortiGate user group definitions and uses established authentication mechanisms such as PAP, CHAP, RADIUS, and LDAP to authenticate dialup clients. You can configure a FortiGate unit to function either as an XAuth server or an XAuth client.If the server or client is attempting a connection using XAuth and the other end is not using XAuth, the failed connection attempts that are logged will not specify XAuth as the reason.

Using the FortiGate unit as an XAuth server

A FortiGate unit can act as an XAuth server for dialup clients. When the Phase 1 negotiation completes, the FortiGate unit challenges the user for a user name and password. It then forwards the user's credentials to an external RADIUS or LDAP server for verification.

If the user records on the RADIUS server have suitably configured Framed-IP-Address fields, you can assign client virtual IP addresses by XAuth instead of from a DHCP address range. See Assigning VIPs by RADIUS user group on page 1.

The authentication protocol to use for XAuth depends on the capabilities of the authentication server and the XAuth client:

- Select **PAP Server** whenever possible.
- You must select **PAP Server** for all implementations of LDAP and some implementations of Microsoft RADIUS.
- Select **Auto Server** when the authentication server supports **CHAP Server** but the XAuth client does not. The FortiGate unit will use PAP to communicate with the XAuth client and CHAP to communicate with the authentication server. You can also use **Auto Server** to allows multiple source interfaces to be defined in an IPsec/IKE policy

Before you begin, create user accounts and user groups to identify the dialup clients that need to access the network behind the FortiGate dialup server. If password protection will be provided through an external RADIUS or LDAP server, you must configure the FortiGate dialup server to forward authentication requests to the authentication server. For information about these topics, see the FortiGate User Authentication Guide.

Authenticating a dialup user group using XAuth settings

1. At the FortiGate dialup server, go to **VPN > IPsec Tunnels** and create the new custom tunnel or edit an existing tunnel.
2. Select **Convert To Custom Tunnel**.
3. Edit **XAUTH**, select the **Type** setting, which determines the type of encryption method to use between the XAuth client, the FortiGate unit and the authentication server. Select one of the following options:
   - **Disabled** — Disables XAuth settings.
   - **PAP Server** — Password Authentication Protocol.
   - **CHAP Server** — Challenge-Handshake Authentication Protocol.
   - **Auto Server** — Use PAP between the XAuth client and the FortiGate unit, and CHAP between the FortiGate unit and the authentication server.
4. From the **User Group** list, select the user group that needs to access the private network behind the FortiGate unit. The group must be added to the FortiGate configuration before it can be selected here. For multiple user groups to be defined in the IPsec/IKE policy, select **Inherit from policy**.
4. Select **OK**.
5. Create as many policies as needed, specifying **Source User(s)** and **Destination Address**.
   For example, one policy could have **user1** have access to **test_local_subnet_1**, while **user2** has access to **test_local_subnet_2**.

As of FortiOS 5.4.1, when XAuth settings are enabled, **Inherit from policy** is only available under **PAP Server** and **CHAP Server**, not **Auto Server**. Because of this, only one user group may be defined for **Auto Server**.

Using the FortiGate unit as an XAuth client

If the FortiGate unit acts as a dialup client, the remote peer, acting as an XAuth server, might require a username and password. You can configure the FortiGate unit as an XAuth client, with its own username and password, which it provides when challenged.

Configuring the FortiGate dialup client as an XAuth client

1. At the FortiGate dialup client, go to **VPN > IPsec Tunnels** and create the new custom tunnel or edit an existing tunnel.
2. Edit the **Phase 1 Proposal** (if it is not available, you may need to click the **Convert to Custom Tunnel** button).
3. Under **XAuth**, select **Enable as Client**.

4. In the **Username** field, type the FortiGate PAP, CHAP, RADIUS, or LDAP user name that the FortiGate XAuth server will compare to its records when the FortiGate XAuth client attempts to connect.
5. In the **Password** field, type the password to associate with the user name.
6. Select **OK**.

Dynamic IPsec route control

You can add a route to a peer destination selector by using the `add-route` option, which is available for all dynamic IPsec Phases 1 and 2, for both policy-based and route-based IPsec VPNs. This option was previously only available when `mode-cfg` was enabled in Phase 1.

The add-route option adds a route to the FortiGate unit's routing information base when the dynamic tunnel is negotiated. You can use the distance and priority options to set the distance and priority of this route. If this results in a route with the lowest distance, it is added to the FortiGate unit's forwarding information base.

You can also enable add-route in any policy-based or route-based Phase 2 configuration that is associated with a dynamic (dialup) Phase 1. In Phase 2, add-route can be enabled, disabled, or set to use the same route as Phase 1.

The add-route feature is enabled by default and is configured in the CLI.

Syntax

Phase 1

```
config vpn ipsec
   edit <name>
      set type dynamic
      set add-route {enable | disable}
   end
end
```

Phase 2

```
config vpn ipsec {phase2 | phase2-interface}
   edit <name>
      set add-route {phase1 | enable | disable}
   end
end
```

Blocking IPsec SA negotiation

For interface-based IPsec, IPsec SA negotiation blocking can only be removed if the peer offers a wildcard selector. If a wildcard selector is offered then the wildcard route will be added to the routing table with the distance/priority value configured in Phase 1 and, if that is the route with the lowest distance, it is installed into the forwarding information base.

In cases where this occurs, it is important to ensure that the distance value configured on Phase 1 is set appropriately.

Phase 2 parameters

This section describes the Phase 2 parameters that are required to establish communication through a VPN.

The following topics are included in this section:

Phase 2 settings

After IPsec VPN Phase 1 negotiations complete successfully, Phase 2 negotiation begins. Phase 2 parameters define the algorithms that the FortiGate unit can use to encrypt and transfer data for the remainder of the session. The basic Phase 2 settings associate IPsec Phase 2 parameters with a Phase 1 configuration.

When defining Phase 2 parameters, you can choose any set of Phase 1 parameters to set up a secure connection and authenticate the remote peer.

For more information on Phase 2 settings in the web-based manager, see IPsec VPN in the web-based manager on page 32.

The information and procedures in this section do not apply to VPN peers that perform negotiations using manual keys.

Phase 2 proposals

In Phase 2, the VPN peer or client and the FortiGate unit exchange keys again to establish a secure communication channel. The Phase 2 Proposal parameters select the encryption and authentication algorithms needed to generate keys for protecting the implementation details of Security Associations (SAs). The keys are generated automatically using a Diffie-Hellman algorithm.

Replay detection

IPsec tunnels can be vulnerable to replay attacks. Replay Detection enables the FortiGate unit to check all IPsec packets to see if they have been received before. If any encrypted packets arrive out of order, the FortiGate unit discards them.

IKE/IPsec Extended Sequence Number (ESN) support

64-bit Extended Sequence numbers (as described in RFC 4303, RFC 4304 as an addition to IKEv1, and RFC 5996 for IKEv2.) are supported for IPsec when Replay Detection is enabled.

Perfect Forward Secrecy (PFS)

By default, Phase 2 keys are derived from the session key created in Phase 1. Perfect Forward Secrecy (PFS) forces a new Diffie-Hellman exchange when the tunnel starts and whenever the Phase 2 keylife expires, causing a new key to be generated each time. This exchange ensures that the keys created in Phase 2 are unrelated to the Phase 1 keys or any other keys generated automatically in Phase 2.

Keylife

The Keylife setting sets a limit on the length of time that a Phase 2 key can be used. The default units are seconds. Alternatively, you can set a limit on the number of kilobytes (KB) of processed data, or both. If you select both, the key expires when either the time has passed or the number of KB have been processed. When the Phase 2 key expires, a new key is generated without interrupting service.

Quick mode selectors

Quick mode selectors determine which IP addresses can perform IKE negotiations to establish a tunnel. By only allowing authorized IP addresses access to the VPN tunnel, the network is more secure.

The default settings are as broad as possible: any IP address or configured address object, using any protocol, on any port.

While the drop down menus for specifying an address also show address groups, the use of address groups may not be supported on a remote endpoint device that is not a FortiGate.

The address groups are at the bottom of the list to make it easy to distinguish between addresses and address groups.

When configuring Quick Mode selector **Source address** and **Destination address**, valid options include IPv4 and IPv6 single addresses, IPv4 subnet, or IPv6 subnet. For more information on IPv6 IPsec VPN, see Overview of IPv6 IPsec support on page 1.

There are some configurations that require specific selectors:

- The VPN peer is a third-party device that uses specific phase2 selectors.
- The FortiGate unit connects as a dialup client to another FortiGate unit, in which case (usually) you must specify a source IP address, IP address range, or subnet. However, this is not required if you are using dynamic routing and mode-cfg.

With FortiOS VPNs, your network has multiple layers of security, with quick mode selectors being an important line of defence.

- Routes guide traffic from one IP address to another.
- Phase 1 and Phase 2 connection settings ensure there is a valid remote end point for the VPN tunnel that agrees on the encryption and parameters.
- Quick mode selectors allow IKE negotiations only for allowed peers.
- Security policies control which IP addresses can connect to the VPN.
- Security policies also control what protocols are allowed over the VPN along with any bandwidth limiting.

FortiOS is limited with IKEv2 selector matching. When using IKEv2 with a named traffic selector, no more than 32 subnets per traffic selector are added, since FortiOS doesn't fully implement the IKEv2 selector matching rules.

The workaround is to use multiple Phase 2s. If the configuration is FGT <-> FGT, then the better alternative is to just use 0.0.0.0 <-> 0.0.0.0 and use the firewall policy for enforcement.

Using the add-route option

Consider using the `add-route` option to add a route to a peer destination selector. Phase 2 includes the option of allowing the `add-route` to automatically match the settings in Phase 1. For more information, refer to Phase 1 parameters on page 46.

Syntax

Phase 2

```
config vpn ipsec {phase2 | phase2-interface}
   edit <name>
      set add-route {phase1 | enable | disable}
   end
end
```

Configuring the Phase 2 parameters

If you are creating a hub-and-spoke configuration or an Internet-browsing configuration, you may have already started defining some of the required Phase 2 parameters. If so, edit the existing definition to complete the configuration.

Specifying the Phase 2 parameters

1. Go to **VPN > IPsec Tunnels** and create the new custom tunnel or edit an existing tunnel.
2. Open the **Phase 2 Selectors** panel (if it is not available, you may need to click the **Convert to Custom Tunnel** button).
3. Enter a **Name** for the Phase 2 configuration, and select a **Phase 1** configuration from the drop-down list.
4. Select **Advanced**.
5. Include the appropriate entries as follows:

Phase 2 Proposal	Select the encryption and authentication algorithms that will be used to change data into encrypted code. Add or delete encryption and authentication algorithms as required. Select a minimum of one and a maximum of three combinations. The remote peer must be configured to use at least one of the proposals that you define. It is invalid to set both **Encryption** and **Authentication** to null.

Encryption	Select a symmetric-key algorithms: **NULL** — Do not use an encryption algorithm. **DES** — Digital Encryption Standard, a 64-bit block algorithm that uses a 56-bit key. **3DES** — Triple-DES; plain text is encrypted three times by three keys. **AES128** — A 128-bit block algorithm that uses a 128-bit key. **AES192** — A 128-bit block algorithm that uses a 192-bit key. **AES256** — A 128-bit block algorithm that uses a 256-bit key. **ChaCha20/Poly1305**— A 128-bit block algorithm that uses a 128-bit key and a symmetric cipher. Only available for IKEv2.
Authentication	You can select either of the following message digests to check the authenticity of messages during an encrypted session: **NULL** — Do not use a message digest. **MD5** — Message Digest 5. **SHA1** — Secure Hash Algorithm 1 - a 160-bit message digest. To specify one combination only, set the **Encryption** and **Authentication** options of the second combination to NULL. To specify a third combination, use the ***Add*** button beside the fields for the second combination. For information regarding NP accelerated offloading of IPsec VPN authentication algorithms, please refer to the Hardware Acceleration handbook chapter.
Enable replay detection	Optionally enable or disable replay detection. Replay attacks occur when an unauthorized party intercepts a series of IPsec packets and replays them back into the tunnel.
Enable perfect forward secrecy (PFS)	Enable or disable PFS. Perfect forward secrecy (PFS) improves security by forcing a new Diffie-Hellman exchange whenever keylife expires.
Diffie-Hellman Group	Select one Diffie-Hellman group (1, 2, 5, 14 through 21, or 27 through 30). The remote peer or dialup client must be configured to use the same group.
Keylife	Select the method for determining when the Phase 2 key expires: **Seconds**, **KBytes**, or **Both**. If you select **Both**, the key expires when either the time has passed or the number of KB have been processed. The range is from 120 to 172800 seconds, or from 5120 to 2147483648 KB.
Autokey Keep Alive	Enable the option if you want the tunnel to remain active when no data is being processed.
Auto-negotiate	Enable the option if you want the tunnel to be automatically renegotiated when the tunnel expires.

DHCP-IPsec	Select **Enable** if the FortiGate unit acts as a dialup server and FortiGate DHCP server or relay will be used to assign VIP addresses to FortiClient dialup clients. The DHCP server or relay parameters must be configured separately. If the FortiGate unit acts as a dialup server and the FortiClient dialup client VIP addresses match the network behind the dialup server, select **Enable** to cause the FortiGate unit to act as a proxy for the dialup clients. This is available only for Phase 2 configurations associated with a dialup Phase 1 configuration. It works only on policy-based VPNs.

Autokey Keep Alive

The Phase 2 SA has a fixed duration. If there is traffic on the VPN as the SA nears expiry, a new SA is negotiated and the VPN switches to the new SA without interruption. If there is no traffic, however, the SA expires (by default) and the VPN tunnel goes down. A new SA will not be generated until there is traffic.

The Autokey Keep Alive option ensures that a new Phase 2 SA is negotiated, even if there is no traffic, so that the VPN tunnel stays up.

Auto-negotiate

By default, the Phase 2 security association (SA) is not negotiated until a peer attempts to send data. The triggering packet and some subsequent packets are dropped until the SA is established. Applications normally resend this data, so there is no loss, but there might be a noticeable delay in response to the user.

If the tunnel goes down, the auto-negotiate feature (when enabled) attempts to re-establish the tunnel. Auto-negotiate initiates the Phase 2 SA negotiation automatically, repeating every five seconds until the SA is established.

Automatically establishing the SA can be important for a dialup peer. It ensures that the VPN tunnel is available for peers at the server end to initiate traffic to the dialup peer. Otherwise, the VPN tunnel does not exist until the dialup peer initiates traffic.

The auto-negotiate feature is available through the Command Line Interface (CLI) via the following commands:

```
config vpn ipsec phase2
   edit <phase2_name>
      set auto-negotiate enable
   end
```

Installing dynamic selectors via auto-negotiate

The IPsec SA connect message generated is used to install dynamic selectors. These selectors can now be installed via the auto-negotiate mechanism. When phase 2 has auto-negotiate enabled, and phase 1 has mesh-selector-type set to **subnet**, a new dynamic selector will be installed for each combination of source and destination subnets. Each dynamic selector will inherit the auto-negotiate option from the template selector and begin SA negotiation. Phase 2 selector sources from dial-up clients will all establish SAs without traffic being initiated from the client subnets to the hub.

DHCP-IPsec

Select this option if the FortiGate unit assigns VIP addresses to FortiClient dialup clients through a DHCP server or relay. This option is available only if the Remote Gateway in the Phase 1 configuration is set to Dialup User and it works only on policy-based VPNs.

With the DHCP-IPsec option, the FortiGate dialup server acts as a proxy for FortiClient dialup clients that have VIP addresses on the subnet of the private network behind the FortiGate unit. In this case, the FortiGate dialup server acts as a proxy on the local private network for the FortiClient dialup client. A host on the network behind the dialup server issues an ARP request, corresponding to the device MAC address of the FortiClient host (when a remote server sends an ARP to the local FortiClient dialup client). The FortiGate unit then answers the ARP request on behalf of the FortiClient host, and forwards the associated traffic to the FortiClient host through the tunnel.

Acting as a proxy prevents the VIP address assigned to the FortiClient dialup client from causing possible ARP broadcast problems — the normal and VIP addresses can confuse some network switches by two addresses having the same MAC address.

IPsec support for ChaCha20/Poly1305 AEAD cipher

In IKEv2, to support RFC 7634, crypto algorithms ChaCha20 and Poly1305 can be used together as a combined mode AEAD cipher (like aes-gcm) in the new `crypto_ftnt` cipher in `cipher_chacha20poly1305.c`.

Syntax

```
config vpn ipsec phase2-interface
   edit <name>
      set phase1name <name>
      set proposal chacha20poly1305
   next
end
```

IPsec support for AES-GCM for IKEv2 Phase 1

In IKEv2, to support RFC 5282, AEAD algorithm AES-GCM is now supported, both 128 and 256-bit variants.

Syntax

```
config vpn ipsec phase2-interface
   edit <name>
      set phase1name <name>
      set proposal [aes128gcm | aes256gcm]
   next
end
```

Defining VPN security policies

This section explains how to specify the source and destination IP addresses of traffic transmitted through an IPsec VPN, and how to define appropriate security policies.

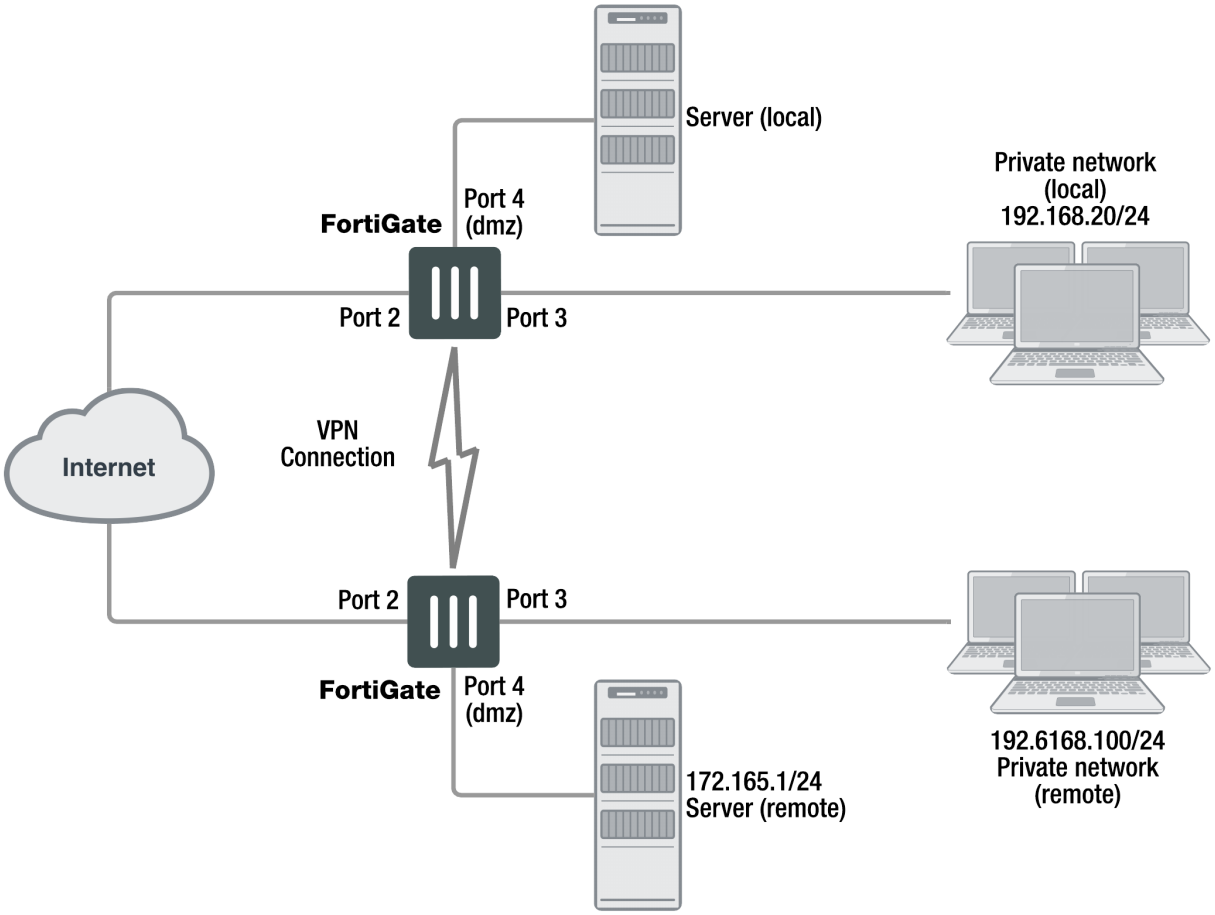
The following topics are included in this section:

Defining policy addresses

A VPN tunnel has two end points. These end points may be VPN peers such as two FortiGate gateways. Encrypted packets are transmitted between the end points. At each end of the VPN tunnel, a VPN peer intercepts encrypted packets, decrypts the packets, and forwards the decrypted IP packets to the intended destination.

You need to define firewall addresses for the private networks behind each peer. You will use these addresses as the source or destination address depending on the security policy.

Example topology for the following policies



In general:

- In a gateway-to-gateway, hub-and-spoke, dynamic DNS, redundant-tunnel, or transparent configuration, you need to define a policy address for the private IP address of the network behind the remote VPN peer (for example, `192.168.10.0/255.255.255.0` or `192.168.10.0/24`).
- In a peer-to-peer configuration, you need to define a policy address for the private IP address of a server or host behind the remote VPN peer (for example, `172.16.5.1/255.255.255.255` or `172.16.5.1/32` or `172.16.5.1`).

For a FortiGate dialup server in a dialup-client or Internet-browsing configuration:

- If you are not using VIP addresses, or if the FortiGate dialup server assigns VIP addresses to FortiClient dialup clients through FortiGate DHCP relay, select the predefined destination address “all” in the security policy to refer to the dialup clients.
- If you assign VIP addresses to FortiClient dialup clients manually, you need to define a policy address for the VIP address assigned to the dialup client (for example, `10.254.254.1/32`), or a subnet address from which the VIP addresses are assigned (for example, `10.254.254.0/24` or `10.254.254.0/255.255.255.0`).
- For a FortiGate dialup client in a dialup-client or Internet-browsing configuration, you need to define a policy address for the private IP address of a host, server, or network behind the FortiGate dialup server.

Defining a security IP address

1. Go to **Policy & Objects > Addresses** and select **Create New**.
2. In the **Name** field, type a descriptive name that represents the network, server(s), or host(s).
3. In **Type**, select **Subnet**.
4. In the **Subnet/IP Range** field, type the corresponding IP address and subnet mask.
   For a subnet you could use the format `172.16.5.0/24` or its equivalent `172.16.5.0/255.255.255.0`. For a server or host it would likely be `172.16.5.1/32`. Alternately you can use an IP address range such as `192.168.10.[80-100]` or `192.168.10.80-192.168.10.100`.
5. Select **OK**.

Defining security policies for policy-based and route-based VPNs

Security policies allow IP traffic to pass between interfaces on a FortiGate unit. You can limit communication to particular traffic by specifying source address and destination addresses. Then only traffic from those addresses will be allowed.

Policy-based and route-based VPNs require different security policies.

- A policy-based VPN requires an IPsec security policy. You specify the interface to the private network, the interface to the remote peer and the VPN tunnel. A single policy can enable traffic inbound, outbound, or in both directions.
- A route-based VPN requires an Accept security policy for each direction. As source and destination interfaces, you specify the interface to the private network and the virtual IPsec interface (Phase 1 configuration) of the VPN. The IPsec interface is the destination interface for the outbound policy and the source interface for the inbound policy. One security policy must be configured for each direction of each VPN interface.

There are examples of security policies for both policy-based and route-based VPNs throughout this guide. See Route-based or policy-based VPN on page 117.

If the security policy, which grants the VPN Connection is limited to certain services, DHCP must be included, otherwise the client won't be able to retrieve a lease from the FortiGate's (IPsec) DHCP server, because the DHCP Request (coming out of the tunnel) will be blocked.

Policy-based VPN

An IPsec security policy enables the transmission and reception of encrypted packets, specifies the permitted direction of VPN traffic, and selects the VPN tunnel. In most cases, a single policy is needed to control both inbound and outbound IP traffic through a VPN tunnel. Be aware of the following considerations below before creating an IPsec security policy.

Allow traffic to be initiated from the remote site

Security policies specify which IP addresses can initiate a tunnel. By default, traffic from the local private network initiates the tunnel. When the **Allow traffic to be initiated form the remote site** option is selected, traffic from a dialup client, or a computer on a remote network, initiates the tunnel. Both can be enabled at the same time for bi-directional initiation of the tunnel.

Outbound and inbound NAT

When a FortiGate unit operates in NAT mode, you can also enable inbound or outbound NAT. Outbound NAT may be performed on outbound encrypted packets or IP packets in order to change their source address before they are sent through the tunnel. Inbound NAT is performed to intercept and decrypt emerging IP packets from the tunnel.

By default, these options are not selected in security policies and can only be set through the CLI. For more information on this, see the “config firewall” chapter of the FortiGate CLI Reference.

Source and destination addresses

Most security policies control outbound IP traffic. A VPN outbound policy usually has a source address originating on the private network behind the local FortiGate unit, and a destination address belonging to a dialup VPN client or a network behind the remote VPN peer. The source address that you choose for the security policy identifies from where outbound cleartext IP packets may originate, and also defines the local IP address or addresses that a remote server or client will be allowed to access through the VPN tunnel. The destination address that you choose identifies where IP packets must be forwarded after they are decrypted at the far end of the tunnel, and determines the IP address or addresses that the local network will be able to access at the far end of the tunnel.

Enabling other policy features

You can fine-tune a policy for services such as HTTP, FTP, and POP3, enable logging, traffic shaping, antivirus protection, web filtering, email filtering, file transfer, email services, and optionally allow connections according to a predefined schedule.

As an option, differentiated services (diffserv or DSCP) for the security policy can be enabled through the CLI. For more information on this feature, see the Traffic Shaping handbook chapter, or the “firewall” chapter of the FortiGate CLI Reference.

Before you begin

Before you define the IPsec policy, you must:

- Define the IP source and destination addresses. See Defining policy addresses on page 72.
- Specify the Phase 1 authentication parameters. See Phase 1 parameters on page 46.
- Specify the Phase 2 parameters. See Phase 2 parameters on page 66.

Defining an IPsec security policy

1. Go to **Policy & Objects > IPv4 Policy**.
2. Select **Create New** and set the following options:

Name	Enter a name for the security policy.
Incoming Interface	Select the local interface to the internal (private) network.
Outgoing Interface	Select the local interface to the external (public) network.
Source	Select the name that corresponds to the local network, server(s), or host(s) from which IP packets may originate.

Destination Address	Select the name that corresponds to the remote network, server(s), or host (s) to which IP packets may be delivered.
Schedule	Keep the default setting (always) unless changes are needed to meet specific requirements.
Service	Keep the default setting (ANY) unless changes are needed to meet your specific requirements.
Action	For the purpose of this configuration, set **Action** to **IPsec**. Doing this will close Firewall / Network Options and open VPN Tunnel options. Select the VPN tunnel of your choice, and select **Allow traffic to be initiated from the remote site**, which will allow traffic from the remote network to initiate the tunnel.

3. You may enable UTM features, and/or event logging, or select advanced settings to authenticate a user group, or shape traffic. For more information, see the Firewall handbook chapter.
4. Select **OK**.
5. Place the policy in the policy list above any other policies having similar source and destination addresses.

Defining multiple IPsec policies for the same tunnel

You must define at least one IPsec policy for each VPN tunnel. If the same remote server or client requires access to more than one network behind a local FortiGate unit, the FortiGate unit must be configured with an IPsec policy for each network. Multiple policies may be required to configure redundant connections to a remote destination or control access to different services at different times.

To ensure a secure connection, the FortiGate unit must evaluate policies with **Action** set to **IPsec** before **ACCEPT** and **DENY**. Because the FortiGate unit reads policies starting at the top of the list, you must move all IPsec policies to the top of the list, and be sure to reorder your multiple IPsec policies that apply to the tunnel so that specific constraints can be evaluated before general constraints.

Adding multiple IPsec policies for the same VPN tunnel can cause conflicts if the policies specify similar source and destination addresses, but have different settings for the same service. When policies overlap in this manner, the system may apply the wrong IPsec policy or the tunnel may fail.

For example, if you create two equivalent IPsec policies for two different tunnels, it does not matter which one comes first in the list of IPsec policies — the system will select the correct policy based on the specified source and destination addresses. If you create two different IPsec policies for the same tunnel (that is, the two policies treat traffic differently depending on the nature of the connection request), you might have to reorder the IPsec policies to ensure that the system selects the correct IPsec policy.

Route-based VPN

When you define a route-based VPN, you create a virtual IPsec interface on the physical interface that connects to the remote peer. You create ordinary Accept security policies to enable traffic between the IPsec interface and the interface that connects to the private network. This makes configuration simpler than for policy-based VPNs, which require IPsec security policies.

Defining security policies for a route-based VPN

1. Go to **Policy & Objects > IPv4 Policy**.
2. Select **Create New** and define an **ACCEPT** security policy to permit communication between the local private network and the private network behind the remote peer. Enter these settings in particular:

Name	Enter a name for the security policy.
Incoming Interface	Select the interface that connects to the private network behind this FortiGate unit.
Outgoing Interface	Select the IPsec Interface you configured.
Source	Select the address name that you defined for the private network behind this FortiGate unit.
Destination Address	Select the address name that you defined for the private network behind the remote peer.
Action	Select **ACCEPT**.
NAT	Disable **NAT**.

To permit the remote client to initiate communication, you need to define a security policy for communication in that direction.

3. Select **Create New** and enter these settings in particular:

Name	Enter a name for the security policy.
Incoming Interface	Select the IPsec Interface you configured.
Outgoing Interface	Select the interface that connects to the private network behind this FortiGate unit.
Source	Select the address name that you defined for the private network behind the remote peer.
Destination Address	Select the address name that you defined for the private network behind this FortiGate unit.
Action	Select **ACCEPT**.
NAT	Disable **NAT**.

Gateway-to-gateway

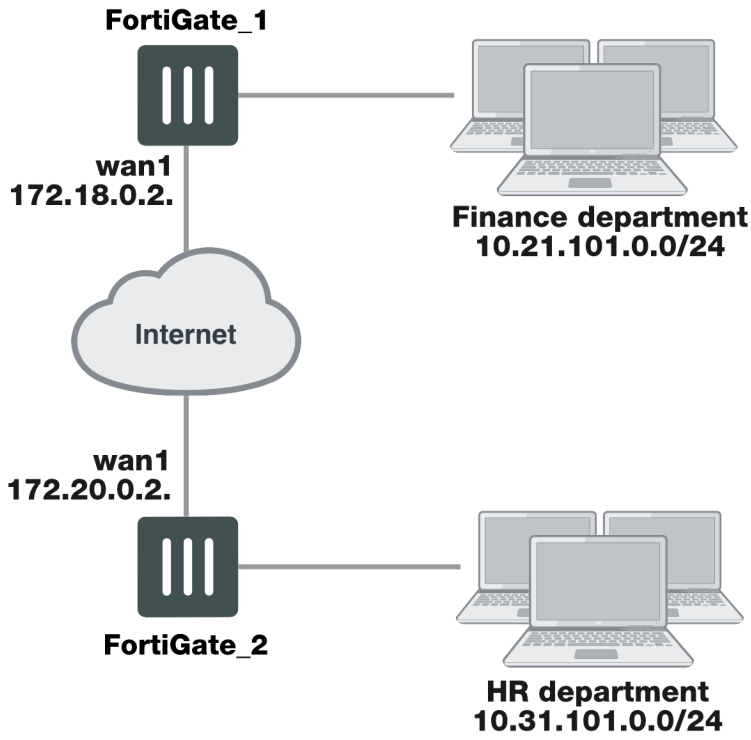
This section explains how to set up a basic gateway-to-gateway (site-to-site) IPsec VPN.

The following topics are included in this section:

Configuration overview

In a gateway-to-gateway configuration, two FortiGate units create a VPN tunnel between two separate private networks. All traffic between the two networks is encrypted and protected by FortiGate security policies.

Example gateway-to-gateway configuration

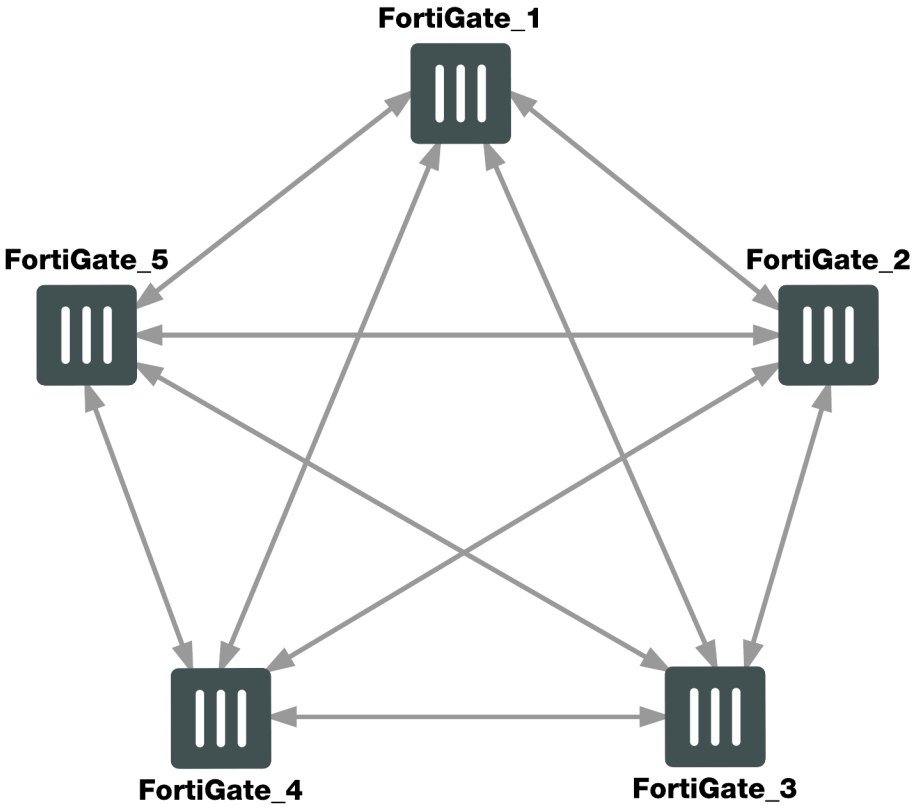


In some cases, computers on the private network behind one VPN peer may (by co-incidence) have IP addresses that are already used by computers on the network behind the other VPN peer. In this type of situation (ambiguous routing), conflicts may occur in one or both of the FortiGate routing tables and traffic destined for the remote network through the tunnel may not be sent. To resolve issues related to ambiguous routing, see Configuration overview on page 78.

In other cases, computers on the private network behind one VPN peer may obtain IP addresses from a local DHCP server. However, unless the local and remote networks use different private network address spaces, unintended ambiguous routing and/or IP-address overlap issues may arise. For a discussion of the related issues, see FortiGate dialup-client configurations on page 1.

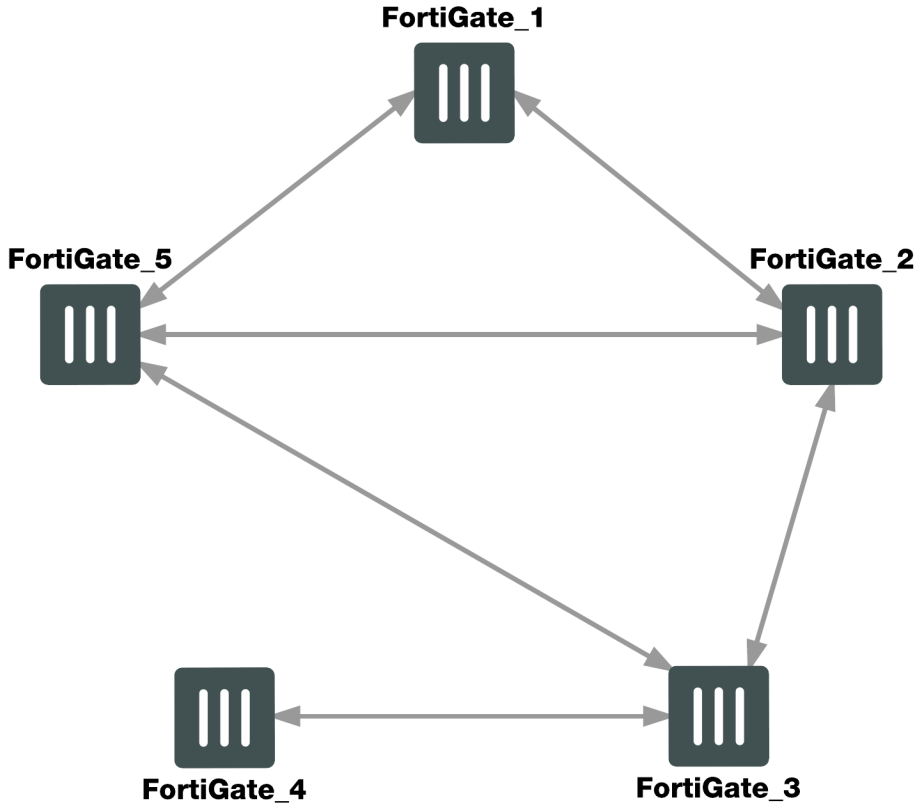
You can set up a fully meshed or partially meshed configuration (see below).

Fully meshed configuration



In a fully meshed network, all VPN peers are connected to each other, with one hop between peers. This topology is the most fault-tolerant: if one peer goes down, the rest of the network is not affected. This topology is difficult to scale because it requires connections between all peers. In addition, unnecessary communication can occur between peers. Best practices dictates a hub-and-spoke configuration instead (see Hub-and-spoke configurations on page 1).

Partially meshed configuration



A partially meshed network is similar to a fully meshed network, but instead of having tunnels between all peers, tunnels are only configured between peers that communicate with each other regularly.

Gateway-to-gateway configuration

The FortiGate units at both ends of the tunnel must be operating in NAT mode and have static public IP addresses.

When a FortiGate unit receives a connection request from a remote VPN peer, it uses IPsec Phase 1 parameters to establish a secure connection and authenticate that VPN peer. Then, if the security policy permits the connection, the FortiGate unit establishes the tunnel using IPsec Phase 2 parameters and applies the IPsec security policy. Key management, authentication, and security services are negotiated dynamically through the IKE protocol.

To support these functions, the following general configuration steps must be performed by both FortiGate units:

- Define the Phase 1 parameters that the FortiGate unit needs to authenticate the remote peer and establish a secure connection.
- Define the Phase 2 parameters that the FortiGate unit needs to create a VPN tunnel with the remote peer.
- Create security policies to control the permitted services and permitted direction of traffic between the IP source and destination addresses.

Configuring Phase 1 and Phase 2 for both peers

This procedure applies to both peers. Repeat the procedure on each FortiGate unit, using the correct IP address for each. You may wish to vary the Phase 1 names but this is optional. Otherwise all steps are the same for each peer.

The Phase 1 configuration defines the parameters that FortiGate_1 will use to authenticate FortiGate_2 and establish a secure connection. For the purposes of this example, a preshared key will be used to authenticate FortiGate_2. The same preshared key must be specified at both FortiGate units.

Before you define the Phase 1 parameters, you need to:

- Reserve a name for the remote gateway.
- Obtain the IP address of the public interface to the remote peer.
- Reserve a unique value for the preshared key.

The key must contain at least 6 printable characters and best practices dictate that it only be known by network administrators. For optimum protection against currently known attacks, the key must have a minimum of 16 randomly chosen alphanumeric characters.

At the local FortiGate unit, define the Phase 1 configuration needed to establish a secure connection with the remote peer. See IPsec VPN in the web-based manager on page 32.

1. Go to **VPN > IPsec Tunnels** and create the new custom tunnel or edit an existing tunnel.
2. Edit the **Phase 1 Proposal** (if it is not available, you may need to click the **Convert to Custom Tunnel** button).
3. Enter the following information, and select **OK**.

Name	Enter `peer_1`. A name to identify the VPN tunnel. This name appears in Phase 2 configurations, security policies and the VPN monitor.
Remote Gateway	Select **Static IP Address**.
IP Address	Enter `172.20.0.2` when configuring FortiGate_1. Enter `172.18.0.2` when configuring FortiGate_2. The IP address of the remote peer public interface.
Local Interface	Select **wan1**.

The basic Phase 2 settings associate IPsec Phase 2 parameters with the Phase 1 configuration and specify the remote end point of the VPN tunnel. Before you define the Phase 2 parameters, you need to reserve a name for the tunnel. See IPsec VPN in the web-based manager on page 32.

1. Open the **Phase 2 Selectors** panel (if it is not available, you may need to click the **Convert to Custom Tunnel** button).
2. Enter a **Name** of `peer_1_p2`.
3. Select **peer_1** from the **Phase 1** drop-down menu.

Creating security policies

Security policies control all IP traffic passing between a source address and a destination address.

An IPsec security policy is needed to allow the transmission of encrypted packets, specify the permitted direction of VPN traffic, and select the VPN tunnel that will be subject to the policy. A single policy is needed to control both inbound and outbound IP traffic through a VPN tunnel.

Before you define security policies, you must first specify the IP source and destination addresses. In a gateway-to-gateway configuration:

- The IP source address corresponds to the private network behind the local FortiGate unit.
- The IP destination address refers to the private network behind the remote VPN peer.

When you are creating security policies, choose one of either route-based or policy-based methods and follow it for both VPN peers. DO NOT configure both route-based and policy-based policies on the same FortiGate unit for the same VPN tunnel.

The configuration of FortiGate_2 is similar to that of FortiGate_1. You must:

- Define the Phase 1 parameters that FortiGate_2 needs to authenticate FortiGate_1 and establish a secure connection.
- Define the Phase 2 parameters that FortiGate_2 needs to create a VPN tunnel with FortiGate_1.
- Create the security policy and define the scope of permitted services between the IP source and destination addresses.

When creating security policies it is good practice to include a comment describing what the policy does.

Creating firewall addresses

Define names for the addresses or address ranges of the private networks that the VPN links. These addresses are used in the security policies that permit communication between the networks.

To define the IP address of the network behind FortiGate_1

1. Go to **Policy & Objects > Addresses** and select **Create New**.
2. Enter the **Name** of `Finance_network`.
3. Select a **Type** of **Subnet**.
4. Enter the **Subnet** of `10.21.101.0/24`.
5. Select **OK**.

To specify the address of the network behind FortiGate_2

1. Go to **Policy & Objects > Addresses** and select **Create New**.
2. Enter the **Name** of `HR_network`.
3. Select a **Type** of **Subnet**.
4. Enter the **Subnet/IP Range** of `10.31.101.0/24`.
5. Select **OK**.

Creating route-based VPN security policies

Define an ACCEPT security policy to permit communications between the source and destination addresses.

To create route-based VPN security policies

1. Go to **Policy & Objects > IPv4 Policy** and select **Create New**
2. Leave the **Policy Type** as **Firewall** and leave the **Policy Subtype** as **Address**.

3. Enter the following, and select **OK**.

Incoming Interface	Select **internal**. The interface that connects to the private network behind this FortiGate unit.
Source Address	Select **Finance_network** when configuring FortiGate_1. Select **HR_network** when configuring FortiGate_2. The address name for the private network behind this FortiGate unit.
Outgoing Interface	Select **peer_1**. The VPN Tunnel (IPsec Interface) you configured earlier.
Destination Address	Select **HR_network** when configuring FortiGate_1. Select **Finance_network** when configuring FortiGate_2. The address name that you defined for the private network behind the remote peer.
Action	Select **ACCEPT**.
Enable NAT	Disable.
Comments	Allow Internal to remote VPN network traffic.

4. Optionally, configure any additional features you may want, such as UTM or traffic shaping.
5. Select **Create New** to create another policy for the other direction.
6. Leave the **Policy Type** as **Firewall** and leave the **Policy Subtype** as **Address**.
7. Enter the following information, and select **OK**.

Incoming Interface	Select **peer_1**. The VPN Tunnel (IPsec Interface) you configured.
Source Address	Select **HR_network** when configuring FortiGate_1. Select **Finance_Network** when configuring FortiGate_2. The address name defined for the private network behind the remote peer.
Outgoing Interface	Select **internal**. The interface that connects to the private network behind this FortiGate unit.

Destination Address	Select **Finance_Network** when configuring FortiGate_1. Select **HR_network** when configuring FortiGate_2. The address name defined for the private network behind this FortiGate unit.
Action	Select **ACCEPT**.
Enable NAT	Disable.
Comments	Allow remote VPN network traffic to Internal.

8. Configure any additional features such as UTM or traffic shaping you may want. (optional).

All network traffic must have a static route to direct its traffic to the proper destination. Without a route, traffic will not flow even if the security policies are configured properly. You may need to create a static route entry for both directions of VPN traffic if your security policies allow bi-directional tunnel initiation.

To configure the route for a route-based VPN:

1. On FortiGate_2, go to **Network > Static Routes** and select **Create New**.
2. Enter the following information, and then select **OK**:

Destination IP / Mask	10.21.101.0/24
Device	FGT2_to_FGT1_Tunnel
Gateway	Leave as default: 0.0.0.0.
Distance (Advanced)	Leave this at its default. If there are other routes on this FortiGate unit, you may need to set the distance on this route so the VPN traffic will use it as the default route. However, this normally happens by default because this route is typically a better match than the generic default route.

Creating policy-based VPN security policy

Define an IPsec security policy to permit communications between the source and destination addresses.

1. Go to **Policy & Objects > IPv4 Policy**.
2. Complete the following:

Incoming Interface	Select **internal**. The interface that connects to the private network behind this FortiGate unit.

Source Address	Select **Finance_network** when configuring FortiGate_1. Select **HR_network** when configuring FortiGate_2. The address name defined for the private network behind this FortiGate unit.
Outgoing Interface	Select **wan1**. The FortiGate unit's public interface.
Destination Address	Select **HR_network** when configuring FortiGate_1. Select **Finance_network** when configuring FortiGate_2.
VPN Tunnel	Select **Use Existing** and select **peer_1** from the **VPN Tunnel** drop-down list. Select **Allow traffic to be initiated from the remote site** to enable traffic from the remote network to initiate the tunnel.
Comments	Bidirectional policy-based VPN policy.

Place VPN policies in the policy list above any other policies having similar source and destination addresses.

Remote Internet browsing for Site-to-Site VPN from the IPsec VPN Wizard

The IPsec VPN Wizard **Policy & Routing** section includes **Internet Access** options to support selecting **Share WAN** and **Force to use remote WAN**:

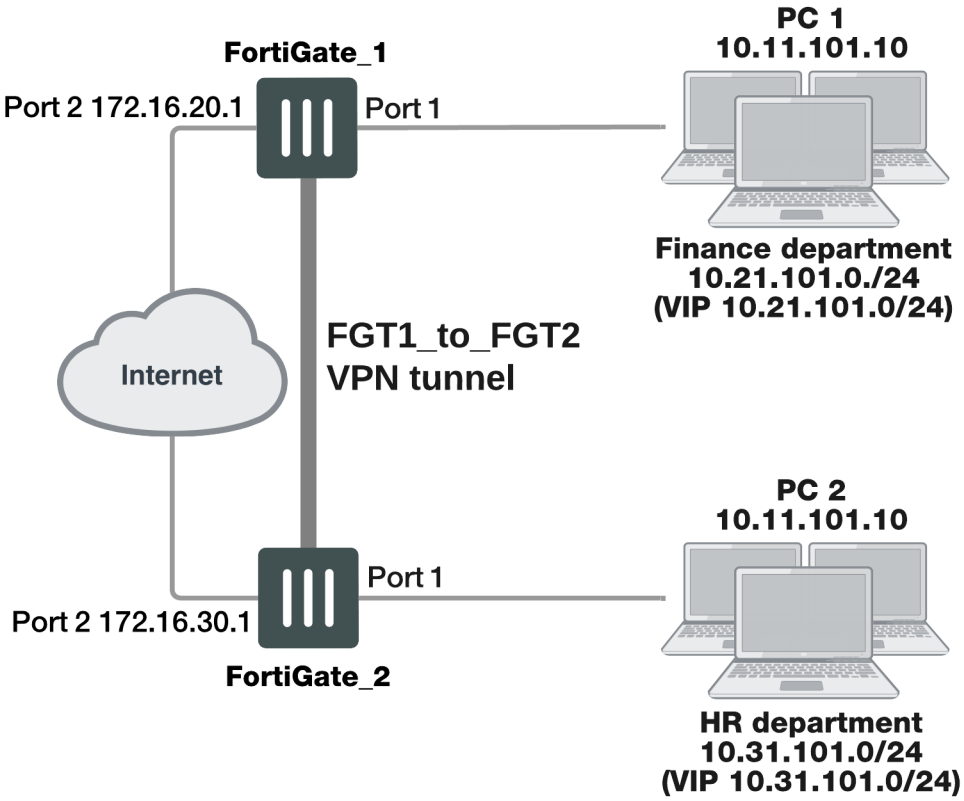
- The **Share WAN** option allows the remote subnet to browse the Internet via this FortiGate.
  When **Share WAN** is selected, a dropdown appears for the user to select the desired **Shared WAN**.
- The **Force to use remote WAN** option will send all Internet browsing traffic to the remote VPN gateway. The remote gateway must be configured with the **Share WAN** option enabled.
  When **Force to use remote WAN** is selected, a **Local Gateway** field appears (since a static route needs to be created to reach the remote gateway, because all other addresses will be routed via the VPN tunnel).

How to work with overlapping subnets

A site-to-site VPN configuration sometimes has the problem that the private subnet addresses at each end are the same. You can resolve this problem by remapping the private addresses using virtual IP addresses (VIP).

VIPs allow computers on those overlapping private subnets to each have another set of IP addresses that can be used without confusion. The FortiGate unit maps the VIP addresses to the original addresses. This means if PC1 starts a session with PC2 at 10.31.101.10, FortiGate_2 directs that session to 10.11.101.10 — the actual IP address of PC2.The figure below demonstrates this — Finance network VIP is 10.21.101.0/24 and the HR network is 10.31.101.0/24.

Overlapped subnets example



Solution for route-based VPN

You need to:

- Configure IPsec Phase 1 and Phase 2 as you usually would for a route-based VPN. In this example, the resulting IPsec interface is named `FGT1_to_FGT2`.
- Configure virtual IP (VIP) mapping:
  - the 10.21.101.0/24 network mapped to the 10.11.101.0/24 network on FortiGate_1
  - the 10.31.101.0/24 network mapped to the 10.11.101.0/24 network on FortiGate_2
- Configure an outgoing security policy with ordinary source NAT on both FortiGates.
- Configure an incoming security policy with the VIP as the destination on both FortiGates.
- Configure a route to the remote private network over the IPsec interface on both FortiGates.

To configure VIP mapping on both FortiGates

1. Go to **Policy & Objects > Virtual IPs** and create a new **Virtual IP**.
2. Enter the following information, and select **OK**:

Name	Enter a name, for example, `my_vip`.
External Interface	Select `FGT1_to_FGT2`. The IPsec interface.

VIP Type	Depending on both FortiGates, select one of the following options: • **IPv4**: If both FortiGates use IPv4 (Static NAT). • **IPv6**: If both FortiGates use IPv6 (Static NAT). • **NAT46**: Maps the IPv4 address into an IPv6 prefix. • **NAT64**: Maps the IPv6 address into an IPv4 prefix.
External IP Address/Range	For the **External IP Address** field enter: `10.21.101.1` when configuring FortiGate_1, or `10.31.101.1` when configuring FortiGate_2.
Mapped IP Address/Range	For the **Mapped IP Address** enter `10.11.101.1`. For the **Range** enter `10.11.101.254`.
Port Forwarding	Disable

3. Repeat this procedure on both FortiGate_1 and FortiGate_2.

To configure the outbound security policy on both FortiGates

1. Go to **Policy & Objects > IPv4 Policy** and select **Create New**.
2. Enter the following information, and select **OK**:

Incoming Interface	Select **Port 1**.
Outgoing Interface	Select **FGT1_to_FGT2**. The IPsec interface.
Source	Select **all**.
Destination Address	Select **all**.
Action	Select **ACCEPT**
NAT	Enable **NAT**.

3. Repeat this procedure on both FortiGate_1 and FortiGate_2.

To configure the inbound security policy on both FortiGates

1. Go to **Policy & Objects > IPv4 Policy** and select **Create New**.
2. Enter the following information, and then select **OK**:

Incoming Interface	Select **FGT1_to_FGT2**.

Outgoing Interface	Select **Port 1**. The IPsec interface.
Source	Select **all**.
Destination Address	Select **my-vip**.
Action	Select **ACCEPT**
NAT	Disable **NAT**.

3. Repeat this procedure on both FortiGate_1 and FortiGate_2.

To configure the static route for both FortiGates

1. Go to **Network > Static Routes** and create a new **Route** (or **IPv6 Route** as necessary).
2. Enter the following information, and then select **OK**:

Destination	Enter a subnet of `10.31.101.0/24` when configuring FortiGate_1. Enter a subnet of `10.21.101.0/24` when configuring FortiGate_2.
Device	Select **FGT1_to_FGT2**.
Gateway	Leave as default: 0.0.0.0.
Administrative Distance	Leave at default (**10**). If you have advanced routing on your network, you may have to change this value.
Advanced Options	If you have advanced routing on your network, enable **Advanced Options** and enter a **Priority**.

Solution for policy-based VPN

As with the route-based solution, users contact hosts at the other end of the VPN using an alternate subnet address. PC1 communicates with PC2 using IP address 10.31.101.10, and PC2 communicates with PC1 using IP address 10.21.101.10.

In this solution however, outbound NAT is used to translate the source address of packets from the 10.11.101.0/24 network to the alternate subnet address that hosts at the other end of the VPN use to reply. Inbound packets from the remote end have their destination addresses translated back to the 10.11.101.0/24 network.

For example, PC1 uses the destination address 10.31.101.10 to contact PC2. Outbound NAT on FortiGate_1 translates the PC1 source address to 10.21.101.10. At the FortiGate_2 end of the tunnel, the outbound NAT configuration translates the destination address to the actual PC2 address of 10.11.101.10. Similarly, PC2 replies to PC1 using destination address 10.21.101.10, with the PC2 source address translated to 10.31.101.10. PC1 and PC2 can communicate over the VPN even though they both have the same IP address.

You need to:

- Configure IPsec Phase 1 as you usually would for a policy-based VPN.
- Configure IPsec Phase 2 with the `use-natip disable` CLI option.
- Define a firewall address for the local private network, 10.11.101.0/24.
- Define a firewall address for the remote private network:
  - Define a firewall address for 10.31.101.0/24 on FortiGate_1
  - Define a firewall address for 10.21.101.0/24 on FortiGate_2
- Configure an outgoing IPsec security policy with outbound NAT to map 10.11.101.0/24 source addresses:
  - To the 10.21.101.0/24 network on FortiGate_1
  - To the 10.31.101.0/24 network on FortiGate_2

To configure IPsec Phase 2 - CLI

```
config vpn ipsec phase2
   edit "FGT1_FGT2_p2"
      set keepalive enable
      set pfs enable
      set phase1name FGT1_to_FGT2
      set proposal 3des-sha1 3des-md5
      set replay enable
      set use-natip disable
   end
```

In this example, your Phase 1 definition is named FGT1_to_FGT2. `use-natip` is set to `disable`, so you can specify the source selector using the **`src-addr-type`**, `src-start-ip` / `src-end-ip` or `src-subnet` keywords. This example leaves these keywords at their default values, which specify the subnet `0.0.0.0/0`.

The `pfs` keyword ensures that perfect forward secrecy (PFS) is used. This ensures that each Phase 2 key created is unrelated to any other keys in use.

To define the local private network firewall address

1. Go to **Policy & Objects > Addresses** and create a new **Address**.
2. Enter the following information and select **OK**.

Category	Set to **Address**.
Name	Enter `vpn-local`. A meaningful name for the local private network.
Type	Set to **IP/Netmask**.
Subnet / IP Range	10.11.101.0 255.255.255.0
Interface	Set to **any**.

To define the remote private network firewall address

1. Go to **Policy & Objects > Addresses** and create a new **Address**.
2. Enter the following information, and select **OK**:

Category	Set to **Address**.

Name	Enter `vpn-remote`. A meaningful name for the remote private network.
Type	Set to **IP/Netmask**.
Subnet / IP Range	`10.31.101.0 255.255.255.0` on FortiGate_1. `10.21.101.0 255.255.255.0` on FortiGate_2.
Interface	Any

To configure the IPsec security policy

In the CLI on FortiGate_1, enter the commands:

```
config firewall policy
   edit 1
      set srcintf "port1"
      set dstintf "port2"
      set srcaddr "vpn-local"
      set dstaddr "vpn-remote"
      set action ipsec
      set schedule "always"
      set service "ANY"
      set inbound enable
      set outbound enable
      set vpntunnel "FGT1_to_FGT2"
      set natoutbound enable
      set natip 10.31.101.0 255.255.255.0
   end
```

Optionally, you can set everything except `natip` in the web-based manager and then use the CLI to set `natip`.

Enter the same commands on FortiGate_2, but set natip be `10.21.101.0 255.255.255.0`.

Testing

The best testing is to look at the packets both as the VPN tunnel is negotiated, and when the tunnel is up.

Determining what the other end of the VPN tunnel is proposing

1. Start a terminal program such as PuTTY and set it to log all output.
   When necessary refer to the logs to locate information when output is verbose.
2. Logon to the FortiGate unit using a super_admin account.
3. Enter the following CLI commands.
4. Display all the possible IKE error types and the number of times they have occurred:

   ```
   diag vpn ike errors
   ```

5. Check for existing debug sessions:

   ```
   diag debug info
   ```

If a debug session is running, to halt it enter:

```
diag debug disable
```

6. Confirm your proposal settings:

```
diag vpn ike config list
```

7. If your proposal settings do not match what you expect, make a change to it and save it to force an update in memory. If that fixes the problem, stop here.
8. List the current vpn filter:

```
diag vpn ike filter
```

9. If all fields are set to any, there are no filters set and all VPN IKE packets will be displayed in the debug output. If your system has only a few VPNs, skip setting the filter.
If your system has many VPN connections this will result in very verbose output and make it very difficult to locate the correct connection attempt.
10. Set the VPN filter to display only information from the destination IP address for example 10.10.10.10:

```
diag vpn ike log-filter dst-addr4 10.10.10.10
```

To add more filter options, enter them one per line as above. Other filter options are:

clear	erase the current filter
dst-addr6	the IPv6 destination address range to filter by
dst-port	the destination port range to filter by
interface	interface that IKE connection is negotiated over
list	display the current filter
name	the phase1 name to filter by
negate	negate the specified filter parameter
src-addr4	the IPv4 source address range to filter by
src-addr6	the IPv6 source address range to filter by
src-port	the source port range to filter by
vd	index of virtual domain. 0 matches all

11. Start debugging:

```
diag debug app ike 255
diag debug enable
```

12. Have the remote end attempt a VPN connection.
If the remote end attempts the connection they become the initiator. This situation makes it easier to debug VPN tunnels because then you have the remote information and all of your local information. by initiate the connection,

you will not see the other end's information.

13. If possible go to the web-based manager on your FortiGate unit, go to the VPN monitor and try to bring the tunnel up.
14. Stop the debug output:

```
diag debug disable
```

15. Go back through the output to determine what proposal information the initiator is using, and how it is different from your VPN P1 proposal settings.

    Things to look for in the debug output of attempted VPN connections are shown below.

Important terms to look for in VPN debug output

`initiator`	Starts the VPN attempt, in the above procedure that is the remote end
`responder`	Answers the initiator's request
`local ID`	In aggressive mode, this is not encrypted
`error no SA proposal chosen`	There was no proposal match — there was no encryption-authentication pair in common, usually occurs after a long list of proposal attempts
`R U THERE` and `R U THERE ack`	dead peer detection (dpd), also known as dead gateway detection — after three failed attempts to contact the remote end it will be declared dead, no farther attempts will be made to contact it
`negotiation result`	lists the proposal settings that were agreed on
`SA_life_soft` and `SA_life_hard`	negotiating a new key, and the key life
`R U THERE`	If you see this, it means Phase 1 was successful
`tunnel up`	the negotiation was successful, the VPN tunnel is operational

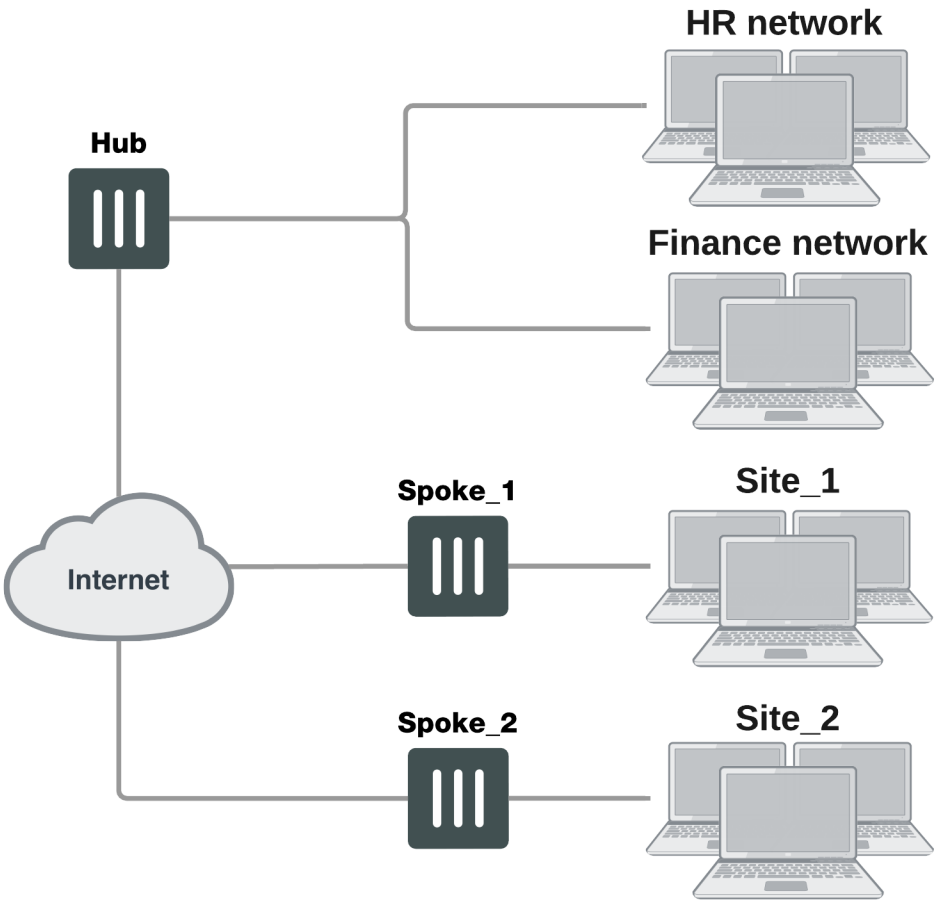
Hub-and-spoke configurations

This section describes how to set up hub-and-spoke IPsec VPNs. The following topics are included in this section:

Configuration overview

In a hub-and-spoke configuration, VPN connections radiate from a central FortiGate unit (the hub) to a number of remote peers (the spokes). Traffic can pass between private networks behind the hub and private networks behind the remote peers. Traffic can also pass between remote peer private networks through the hub.

Example hub-and-spoke configuration



The actual implementation varies in complexity depending on:

- Whether the spokes are statically or dynamically addressed
- The addressing scheme of the protected subnets
- How peers are authenticated

This guide discusses the issues involved in configuring a hub-and-spoke VPN and provides some basic configuration examples.

Hub-and-spoke infrastructure requirements

- The FortiGate hub must be operating in NAT mode and have a static public IP address.
- Spokes may have static IP addresses, dynamic IP addresses (see FortiGate dialup-client configurations on page 137), or static domain names and dynamic IP addresses (see Dynamic DNS configuration on page 115).

Spoke gateway addressing

The public IP address of the spoke is the VPN remote gateway as seen from the hub. Statically addressed spokes each require a separate VPN Phase 1 configuration on the hub. When there are many spokes, this becomes rather cumbersome.

Using dynamic addressing for spokes simplifies the VPN configuration because then the hub requires only a single Phase 1 configuration with “dialup user” as the remote gateway. You can use this configuration even if the remote peers have static IP addresses. A remote peer can establish a VPN connection regardless of its IP address if its traffic selectors match and it can authenticate to the hub. See Configuration overview on page 94 for an example of this configuration.

Protected networks addressing

The addresses of the protected networks are needed to configure destination selectors and sometimes for security policies and static routes. The larger the number of spokes, the more addresses there are to manage. You can

- Assign spoke subnets as part of a larger subnet, usually on a new network

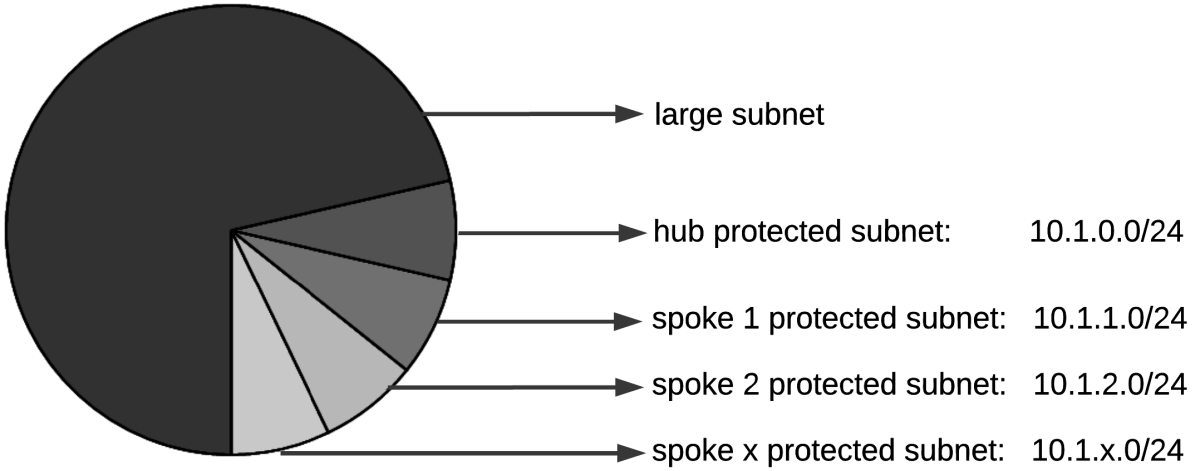
  or

- Create address groups that contain all of the needed addresses

Using aggregated subnets

If you are creating a new network, where subnet IP addresses are not already assigned, you can simplify the VPN configuration by assigning spoke subnets that are part of a large subnet.

Aggregated subnets



All spokes use the large subnet address, 10.1.0.0/16 for example, as:

- The IPsec destination selector
- The destination of the security policy from the private subnet to the VPN (required for policy-based VPN, optional for route-based VPN)
- The destination of the static route to the VPN (route-based)

Each spoke uses the address of its own protected subnet as the IPsec source selector and as the source address in its VPN security policy. The remote gateway is the public IP address of the hub FortiGate unit.

Using an address group

If you want to create a hub-and-spoke VPN between existing private networks, the subnet addressing usually does not fit the aggregated subnet model discussed earlier. All of the spokes and the hub will need to include the addresses of all the protected networks in their configuration.

On FortiGate units, you can define a named firewall address for each of the remote protected networks and add these addresses to a firewall address group. For a policy-based VPN, you can then use this address group as the destination of the VPN security policy.

For a route-based VPN, the destination of the VPN security policy can be set to All. You need to specify appropriate routes for each of the remote subnets.

Authentication

Authentication is by a common pre-shared key or by certificates. For simplicity, the examples in this chapter assume that all spokes use the same pre-shared key.

Configure the hub

At the FortiGate unit that acts as the hub, you need to:

- Configure the VPN to each spoke
- Configure communication between spokes

You configure communication between spokes differently for a policy-based VPN than for a route-based VPN. For a policy-based VPN, you configure a VPN concentrator. For a route-based VPN, you must either define security policies or group the IPsec interfaces into a zone.

Define the hub-spoke VPNs

Perform these steps at the FortiGate unit that will act as the hub. Although this procedure assumes that the spokes are all FortiGate units, a spoke could also be VPN client software, such as FortiClient Endpoint Security.

Configuring the VPN hub

1. At the hub, define the Phase 1 configuration for each spoke. See Phase 1 parameters on page 46. Enter these settings in particular:

Name	Enter a name to identify the VPN in Phase 2 configurations, security policies and the VPN monitor.
Remote Gateway	The remote gateway is the other end of the VPN tunnel. There are three options: **Static IP Address** — Enter the spoke's public **IP Address**. You will need to create a Phase 1 configuration for each spoke. Either the hub or the spoke can establish the VPN connection. **Dialup User** — No additional information is needed. The hub accepts connections from peers with appropriate encryption and authentication settings. Only one Phase 1 configuration is needed for multiple dialup spokes. Only the spoke can establish the VPN tunnel. **Dynamic DNS** — If the spoke subscribes to a dynamic DNS service, enter the spoke's **Dynamic DNS** domain name. Either the hub or the spoke can establish the VPN connection. For more information, see Dynamic DNS configuration on page 1.
Local Interface	Select the FortiGate interface that connects to the remote gateway. This is usually the FortiGate unit's public interface.

2. Define the Phase 2 parameters needed to create a VPN tunnel with each spoke. See Phase 2 parameters on page 66. Enter these settings in particular:

Name	Enter a name to identify this spoke Phase 2 configuration.
Phase 1	Select the name of the Phase 1 configuration that you defined for this spoke.

IPsec VPN in ADVPN hub-and-spoke

IPsec VPN traffic is allowed through a tunnel between an ADVPN hub-and-spoke.

CLI syntax:

```
config vpn ipsec phase1-interface
   edit "int-fgtb"
      ...
      set auto-discovery-sender [enable | disable]
      set auto-discovery-receiver [enable | disable]
      set auto-discovery-forwarder [enable | disable]
      ...
   next
end
config vpn ipsec phase2-interface
   edit "int-fgtb"
      ...
      set auto-discovery-sender phase1 [enable | disable]
      ...
   next
end
```

Define the hub-spoke security policies

1. Define a name for the address of the private network behind the hub. For more information, see Defining policy addresses on page 1.
2. Define names for the addresses or address ranges of the private networks behind the spokes. For more information, see Defining policy addresses on page 1.
3. Define the VPN concentrator. See To define the VPN concentrator on page 99.
4. Define security policies to permit communication between the hub and the spokes. For more information, see Defining VPN security policies on page 1.

Route-based VPN security policies

Define ACCEPT security policies to permit communications between the hub and the spoke. You need one policy for each direction.

Adding policies

1. Go to **Policy & Objects > IPv4 Policy** and select **Create New**.
2. Leave the **Policy Type** as **Firewall** and leave the **Policy Subtype** as **Address**.
3. Enter these settings in particular:

Incoming Interface	Select the VPN Tunnel (IPsec Interface) you configured in Step 1.
Source Address	Select the address name you defined in Step 2 for the private network behind the spoke FortiGate unit.
Outgoing Interface	Select the hub's interface to the internal (private) network.
Destination Address	Select the source address that you defined in Step 1.
Action	Select **ACCEPT**.
Enable NAT	Enable.

Incoming Interface	Select the VPN Tunnel (IPsec Interface) you configured inStep 1.
Source Address	Select the address name you defined in Step 2 for the private network behind the spoke FortiGate units.
Outgoing Interface	Select the source address that you defined in Step 1.
Destination Address	Select the hub's interface to the internal (private) network.
Action	Select **ACCEPT**.
Enable NAT	Enable.

Policy-based VPN security policy

Define an IPsec security policy to permit communications between the hub and the spoke.

Adding policies

1. Go to **Policy & Objects > IPv4 Policy** and select **Create New**.
2. Enter these settings in particular:

Incoming Interface	Select the hub's interface to the internal (private) network.
Source Address	Select the source address that you defined in Step 1.
Outgoing Interface	Select the hub's public network interface.
Destination Address	Select the address name you defined in Step 2 for the private network behind the spoke FortiGate unit.
VPN Tunnel	Select **Use Existing** and select the name of the Phase 1 configuration that you created for the spoke in Step 1. Select **Allow traffic to be initiated from the remote site** to enable traffic from the remote network to initiate the tunnel.

In the policy list, arrange the policies in the following order:

- IPsec policies that control traffic between the hub and the spokes first
- The default security policy last

Configuring communication between spokes (policy-based VPN)

For a policy-based hub-and-spoke VPN, you define a concentrator to enable communication between the spokes.

To define the VPN concentrator

1. At the hub, go to **VPN > IPsec Concentrator** and select **Create New**.
2. In the **Concentrator Name** field, type a name to identify the concentrator.
3. From the **Available Tunnels** list, select a VPN tunnel and then select the right-pointing arrow.

4. Repeat Step 3 until all of the tunnels associated with the spokes are included in the concentrator.
5. Select **OK**.

Configuring communication between spokes (route-based VPN)

For a route-based hub-and-spoke VPN, there are several ways you can enable communication between the spokes:

- Put all of the IPsec interfaces into a zone and enable intra-zone traffic. This eliminates the need for any security policy for the VPN, but you cannot apply UTM features to scan the traffic for security threats.
- Put all of the IPsec interfaces into a zone and create a single zone-to-zone security policy
- Create a security policy for each pair of spokes that are allowed to communicate with each other. The number of policies required increases rapidly as the number of spokes increases.

Using a zone as a concentrator

A simple way to provide communication among all of the spokes is to create a zone and allow intra-zone communication. You cannot apply UTM features using this method.

1. Go to **Network > Interfaces**.
2. Select the down-arrow on the **Create New** button and select **Zone**.
3. In the **Zone Name** field, enter a name, such as `Our_VPN_zone`.
4. Clear **Block intra-zone traffic**.
5. In the **Interface Members** list, select the IPsec interfaces that are part of your VPN.
6. Select **OK**.

Using a zone with a policy as a concentrator

If you put all of the hub IPsec interfaces involved in the VPN into a zone, you can enable communication among all of the spokes and apply UTM features with just one security policy.

Creating a zone for the VPN

1. Go to **Network > Interfaces**.
2. Select the down-arrow on the **Create New** button and select **Zone**.
3. In the **Zone Name** field, enter a name, such as `Our_VPN_zone`.
4. Select **Block intra-zone traffic**.
5. In the **Interface Members** list, select the IPsec interfaces that are part of your VPN.
6. Select **OK**.

Creating a security policy for the zone

1. Go to **Policy & Objects > IPv4 Policy** and select **Create New**.
2. Leave the **Policy Type** as **Firewall** and leave the **Policy Subtype** as **Address**.
3. Enter the settings: and select **OK**.

Incoming Interface	Select the zone you created for your VPN.

Source Address	Select ***All***.
Outgoing Interface	Select the zone you created for your VPN.
Destination Address	Select ***All***.
Action	Select **ACCEPT**.
Enable NAT	Enable.

Using security policies as a concentrator

To enable communication between two spokes, you need to define an ACCEPT security policy for them. To allow either spoke to initiate communication, you must create a policy for each direction. This procedure describes a security policy for communication from Spoke 1 to Spoke 2. Others are similar.

1. Define names for the addresses or address ranges of the private networks behind each spoke. For more information, see Defining policy addresses on page 1.
2. Go to **Policy & Objects > IPv4 Policy** and select **Create New**.
3. Leave the **Policy Type** as **Firewall** and leave the **Policy Subtype** as **Address**.
4. Enter the settings and select **OK**.

Incoming Interface	Select the IPsec interface that connects to Spoke 1.
Source Address	Select the address of the private network behind Spoke 1.
Outgoing Interface	Select the IPsec interface that connects to Spoke 2.
Destination Address	Select the address of the private network behind Spoke 2.
Action	Select **ACCEPT**.
Enable NAT	Enable.

Configure the spokes

Although this procedure assumes that the spokes are all FortiGate units, a spoke could also be VPN client software, such as FortiClient Endpoint Security.

Perform these steps at each FortiGate unit that will act as a spoke.

Creating the Phase 1 and phase_2 configurations

1. At the spoke, define the Phase 1 parameters that the spoke will use to establish a secure connection with the hub. See Phase 1 parameters on page 46. Enter these settings:

Remote Gateway	Select **Static IP Address**.
IP Address	Type the IP address of the interface that connects to the hub.

2. Create the Phase 2 tunnel definition. See Phase 2 parameters on page 66. Select the set of Phase 1 parameters that you defined for the hub. You can select the name of the hub from the **Static IP Address** part of the list.

Configuring security policies for hub-to-spoke communication

1. Create an address for this spoke. See Defining policy addresses on page 1. Enter the IP address and netmask of the private network behind the spoke.
2. Create an address to represent the hub. See Defining policy addresses on page 1. Enter the IP address and netmask of the private network behind the hub.
3. Define the security policy to enable communication with the hub.

Route-based VPN security policy

Define two security policies to permit communications to and from the hub.

1. Go to **Policy & Objects > IPv4 Policy** and select **Create New**.
2. Leave the **Policy Type** as **Firewall** and leave the **Policy Subtype** as **Address**.
3. Enter these settings:

Incoming Interface	Select the virtual IPsec interface you created.
Source Address	Select the hub address you defined in Step 1.
Outgoing Interface	Select the spoke's interface to the internal (private) network.
Destination Address	Select the spoke addresses you defined in Step 2.
Action	Select **ACCEPT**.
Enable NAT	Enable

Incoming Interface	Select the spoke's interface to the internal (private) network.
Source Address	Select the spoke address you defined in Step 1.
Outgoing Interface	Select the virtual IPsec interface you created.
Destination Address	Select the hub destination addresses you defined in Step 2.
Action	Select **ACCEPT**.
Enable NAT	Enable

Policy-based VPN security policy

Define an IPsec security policy to permit communications with the hub. See Defining VPN security policies on page 1.

1. Go to **Policy & Objects > IPv4 Policy** and select **Create New**.
2. Enter these settings in particular:

Incoming Interface	Select the spoke's interface to the internal (private) network.
Source Address	Select the spoke address you defined in Step 1.
Outgoing Interface	Select the spoke's interface to the external (public) network.
Destination Address	Select the hub address you defined in Step 2.
VPN Tunnel	Select **Use Existing** and select the name of the Phase 1 configuration you defined. Select **Allow traffic to be initiated from the remote site** to enable traffic from the remote network to initiate the tunnel.

Configuring security policies for spoke-to-spoke communication

Each spoke requires security policies to enable communication with the other spokes. Instead of creating separate security policies for each spoke, you can create an address group that contains the addresses of the networks behind the other spokes. The security policy then applies to all of the spokes in the group.

1. Define destination addresses to represent the networks behind each of the other spokes. Add these addresses to an address group.
2. Define the security policy to enable communication between this spoke and the spokes in the address group you created.

Policy-based VPN security policy

Define an IPsec security policy to permit communications with the other spokes. See Defining VPN security policies on page 1. Enter these settings in particular:

Route-based VPN security policy

Define two security policies to permit communications to and from the other spokes.

1. Go to **Policy & Objects > IPv4 Policy** and select **Create New**.
2. Leave the **Policy Type** as **Firewall** and leave the **Policy Subtype** as **Address**.
3. Enter these settings in particular:

Incoming Interface	Select the virtual IPsec interface you created.
Source Address	Select the spoke address group you defined in Step "Configure the spokes " on page 101.
Outgoing Interface	Select the spoke's interface to the internal (private) network.
Destination Address	Select this spoke's address name.
Action	Select **ACCEPT**.
Enable NAT	Enable

4. Select **Create New**, leave the **Policy Type** as **Firewall** and leave the **Policy Subtype** as **Address**, and enter these settings:

Incoming Interface	Select the spoke's interface to the internal (private) network.
Source Address	Select this spoke's address name.
Outgoing Interface	Select the virtual IPsec interface you created.
Destination Address	Select the spoke address group you defined in Step 1.
Action	Select **ACCEPT**.
Enable NAT	Enable

Policy-based VPN security policy

1. Go to **Policy & Objects > IPv4 Policy** and select **Create New**.
2. Enter the following:

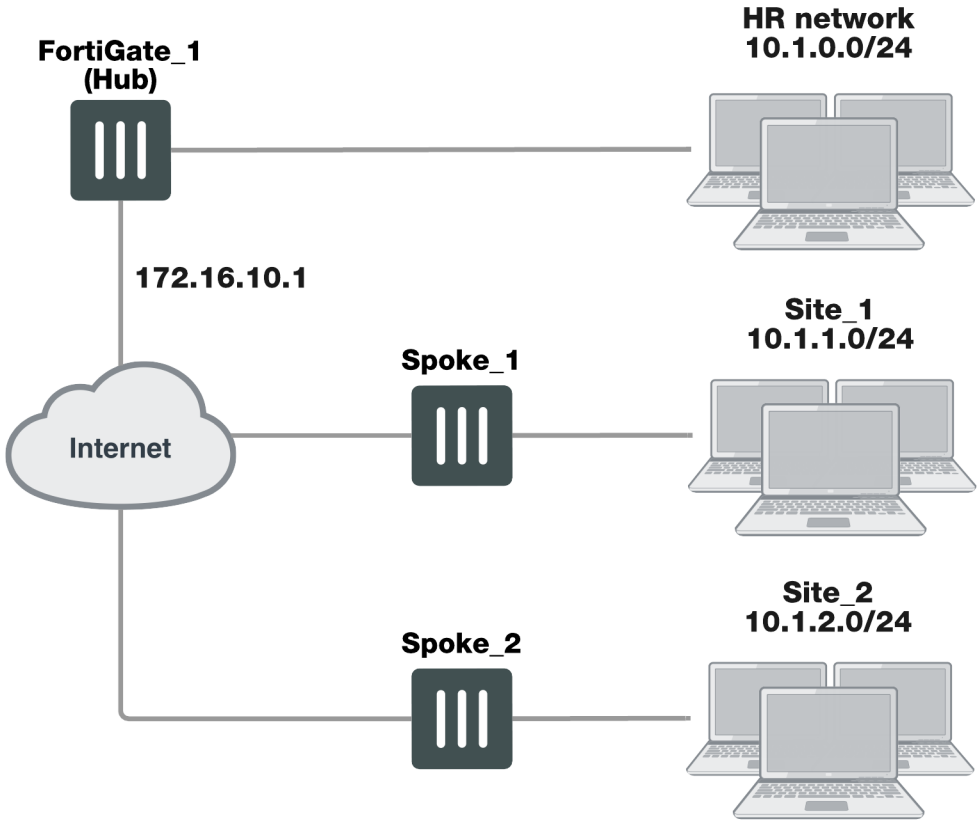
Incoming Interface	Select this spoke's internal (private) network interface.
Source Address	Select this spoke's source address.
Outgoing Interface	Select the spoke's interface to the external (public) network.
Destination Address	Select the spoke address group you defined in Step 1.
VPN Tunnel	Select **Use Existing** and select the name of the Phase 1 configuration you defined. Select **Allow traffic to be initiated from the remote site** to enable traffic from the remote network to initiate the tunnel.

Place this policy or policies in the policy list above any other policies having similar source and destination addresses.

Dynamic spokes configuration example

This example demonstrates how to set up a basic route-based hub-and-spoke IPsec VPN that uses preshared keys to authenticate VPN peers.

Example hub-and-spoke configuration



In the example configuration, the protected networks 10.1.0.0/24, 10.1.1.0/24 and 10.1.2.0/24 are all part of the larger subnet 10.1.0.0/16. The steps for setting up the example hub-and-spoke configuration create a VPN among Site 1, Site 2, and the HR Network.

The spokes are dialup. Their addresses are not part of the configuration on the hub, so only one spoke definition is required no matter the number of spokes. For simplicity, only two spokes are shown.

In an ADVPN topology, any two pair of peers can create a shortcut, as long as one of the devices is not behind NAT.

The on-the-wire format of the ADVPN messages use TLV encoding. Because of this, this feature is not compatible with any previous ADVPN builds.

Configure the hub (FortiGate_1)

The Phase 1 configuration defines the parameters that FortiGate_1 will use to authenticate spokes and establish secure connections.

For the purposes of this example, one preshared key will be used to authenticate all of the spokes. Each key must contain at least 6 printable characters and best practices dictates that it only be known by network administrators. For optimum protection against currently known attacks, each key must consist of a minimum of 16 randomly chosen alphanumeric characters.

Define the IPsec configuration

1. At FortiGate_1, go to **VPN > IPsec Tunnels** and create the new custom tunnel or edit an existing tunnel.
2. Edit the **Phase 1 Proposal** (if it is not available, you may need to click the **Convert to Custom Tunnel** button). Define the Phase 1 parameters that the hub will use to establish a secure connection to the spokes.

Name	Enter a name (for example, `toSpokes`).
Remote Gateway	Dialup user
Local Interface	External
Mode	Main
Authentication Method	Preshared Key
Pre-shared Key	Enter the preshared key.
Peer Options	Any peer ID

The basic Phase 2 settings associate IPsec Phase 2 parameters with the Phase 1 configuration and specify the remote end points of the VPN tunnels.

3. Open the **Phase 2 Selectors** panel (if it is not available, you may need to click the **Convert to Custom Tunnel** button).
4. Enter the following information, and select **OK**:

Name	Enter a name for the Phase 2 definition (for example, `toSpokes_ph2`).
Phase 1	Select the Phase 1 configuration that you defined previously (for example, `toSpokes`).

Define the security policies

security policies control all IP traffic passing between a source address and a destination address. For a route-based VPN, the policies are simpler than for a policy-based VPN. Instead of an IPSEC policy, you use an ACCEPT policy with the virtual IPsec interface as the external interface.

Before you define security policies, you must first define firewall addresses to use in those policies. You need addresses for:

- The HR network behind FortiGate_1
- The aggregate subnet address for the protected networks

Defining the IP address of the HR network behind FortiGate_1

1. Go to **Policy & Objects > Addresses**.
2. Select **Create New**, enter the following information, and select **OK**:

Name	Enter an address name (for example, `HR_Network`).
Type	Subnet

Subnet/IP Range	Enter the IP address of the HR network behind FortiGate_1 (for example, `10.1.0.0/24`).

Specifying the IP address the aggregate protected subnet

1. Go to **Policy & Objects > Addresses**.
2. Select **Create New**, enter the following information, and select **OK**:

Address Name	Enter an address name (for example, `Spoke_net`).
Type	Subnet
Subnet/IP Range	Enter the IP address of the aggregate protected network, `10.1.0.0/16`

Defining the security policy for traffic from the hub to the spokes

1. Go to **Policy & Objects > IPv4 Policy** and select **Create New**,
2. Leave the **Policy Type** as **Firewall** and leave the **Policy Subtype** as **Address**.
3. Enter the following information, and select **OK**:

Incoming Interface	Select the interface to the HR network, **port 1**.
Source Address	`Select` **HR_Network**.
Outgoing Interface	Select the virtual IPsec interface that connects to the spokes, `toSpokes`.
Destination Address	`Select` **Spoke_net**.
Action	Select **ACCEPT**.

Place the policy in the policy list above any other policies having similar source and destination addresses.

Configure communication between spokes

Spokes communicate with each other through the hub. You need to configure the hub to allow this communication. An easy way to do this is to create a zone containing the virtual IPsec interfaces even if there is only one, and create a zone-to-zone security policy.

1. Go to **Network > Interfaces**.
2. Select the down-arrow on the **Create New** button and select **Zone**.
3. In the **Zone Name** field, enter a name, such as `Our_VPN_zone`.
4. Select **Block intra-zone traffic**.
   You could enable intra-zone traffic and then you would not need to create a security policy. But, you would not be able to apply UTM features.
5. In **Interface Members**, select the virtual IPsec interface, `toSpokes`.
6. Select **OK**.

Creating a security policy for the zone

1. Go to **Policy & Objects > IPv4 Policy** and select **Create New**.
2. Leave the **Policy Type** as **Firewall** and leave the **Policy Subtype** as **Address**.

3. Enter these settings:

Incoming Interface	Select `Our_VPN_zone`.
Source Address	Select **All**.
Outgoing Interface	Select `Our_VPN_zone`.
Destination Address	Select **All**.
Action	Select **ACCEPT**.
Enable NAT	Enable.

4. Select **OK**.

Configure the spokes

In this example, all spokes have nearly identical configuration, requiring the following:

- Phase 1 authentication parameters to initiate a connection with the hub.
- Phase 2 tunnel creation parameters to establish a VPN tunnel with the hub.
- A source address that represents the network behind the spoke. This is the only part of the configuration that is different for each spoke.
- A destination address that represents the aggregate protected network.
- A security policy to ena.ble communications between the spoke and the aggregate protected network

Define the IPsec configuration

At each spoke, create the following configuration.

1. At the spoke, go to **VPN > IPsec Tunnels** and create the new custom tunnel or edit an existing tunnel.
2. Edit the **Phase 1 Proposal** (if it is not available, you may need to click the **Convert to Custom Tunnel** button). Enter the following information:

Name	Type a name, for example, `toHub`.
Remote Gateway	Select **Static IP Address**.
IP Address	Enter `172.16.10.1`.
Local Interface	Select **Port2**.
Mode	Main
Authentication Method	Preshared Key
Pre-shared Key	Enter the preshared key. The value must be identical to the preshared key that you specified previously in the FortiGate_1 configuration
Peer Options	Select **Any peer ID**.

1. Open the **Phase 2 Selectors** panel (if it is not available, you may need to click the **Convert to Custom Tunnel** button).
2. Enter the following information and select **OK**:

Name	Enter a name for the tunnel, for example, `toHub_ph2`.
Phase 1	Select the name of the Phase 1 configuration that you defined previously, for example, `toHub`.
Advanced	Select to show the following **Quick Mode Selector** settings.
Source	Enter the address of the protected network at this spoke. For spoke_1, this is `10.1.1.0/24`. For spoke_2, this is `10.1.2.0/24`.
Destination	Enter the aggregate protected subnet address, `10.1.0.0/16`.

Define the security policies

You need to define firewall addresses for the spokes and the aggregate protected network and then create a security policy to enable communication between them.

Defining the IP address of the network behind the spoke

1. Go to **Policy & Objects > Addresses**.
2. Select **Create New** and enter the following information:

Address Name	Enter an address name, for example `LocalNet`.
Type	Subnet
Subnet/IP Range	Enter the IP address of the private network behind the spoke. For spoke_1, this is `10.1.1.0/24`. For spoke_2, this is `10.1.2.0/24`.

Specifying the IP address of the aggregate protected network

1. Go to **Policy & Objects > Addresses**.
2. Select **Create New** and enter the following information:

Address Name	Enter an address name, for example, `Spoke_net`.
Type	Subnet
Subnet/IP Range	Enter the IP address of the aggregate protected network, `10.1.0.0/16`.

Defining the security policy

1. Go to **Policy & Objects > IPv4 Policy** and select **Create New**.
2. Leave the **Policy Type** as **Firewall** and leave the **Policy Subtype** as **Address**.

3. Enter the following information:

Incoming Interface	Select the virtual IPsec interface, `toHub`.
Source Address	Select the aggregate protected network address `Spoke_net`.
Outgoing Interface	Select the interface to the internal (private) network, `port1`.
Destination Address	Select the address for this spoke's protected network `LocalNet`.
Action	Select **ACCEPT**.

4. Select **Create New**.
5. Leave the **Policy Type** as **Firewall** and leave the **Policy Subtype** as **Address**.
6. Enter the following information, and select **OK**:

Incoming Interface	Select the interface to the internal private network, `port1`.
Source Address	Select the address for this spoke's protected network, `LocalNet`.
Outgoing Interface	Select the virtual IPsec interface, `toHub`.
Destination Address	Select the aggregate protected network address, `Spoke_net`.
Action	Select **ACCEPT**.

Place these policies in the policy list above any other policies having similar source and destination addresses.

One-Click VPN (OCVPN)

One-Click VPN (OCVPN) is a cloud-based solution that greatly simplifies the provisioning and configuration of IPsec VPN. The administrator enables OCVPN with a single click, adds the required subnets, and then the configuration is complete. The OCVPN updates each FortiGate automatically as devices join/leave the VPN, as subnets are added/removed, when dynamic external IPs change (e.g. DHCP/PPPoE), and when WAN interface bindings change (as in the dual WAN redundancy case).

Configuration changes and events are automatically propagated across participating nodes without user intervention, so in a sense, the VPN manages itself as a unit with only bare minimum user input. The user specifies which subnets to participate in the VPN. Everything else happens transparently to the user.

After registering devices with FortiCare, devices use SSL to register local subnets with the OCVPN cloud service at https://productapi.fortinet.com. The WAN IP is determined automatically (devices must use a publicly routed external WAN IP address) and the gateway IP address and participating subnets are uploaded to a cloud repository that collects and stores the information in each customer's FortiCare account.

The following limitations apply to FortiOS OCVPN:

- The FortiGate must be registered with a valid FortiCare Support license.
- Only full-mesh VPN configurations using PSK cryptography are supported.
- Public IPs must be used (FortiGates behind NAT cannot participate).
- Non-root VDOMs and FortiGate VMs are not supported.
- Up to 16 nodes can be added to the OCVPN cloud, each with a maximum of 16 subnets.

OCVPN support for High Availability (HA)

As of 6.0.2, HA-enabled devices are now supported by OCVPN.

Prior to establishing the HA cluster, if OCVPN is in use then both devices should be registered to the OCVPN cloud service. During failover, the old serial number is withdrawn and a new serial number (and VPN) is added, to account for the change in status.

General configuration

If FortiCare Support is registered on the FortiGate, you can configure OCVPN in FortiOS under **VPN > One-Click VPN Settings**.

Once enabled, you can add the relevant **Subnets**, as well as view any **Cloud Members** currently participating in the cloud-serviced VPN (you may need to **Refresh** the **Cloud Members** table).

If you wish to change the polling interval, you must do so in the CLI Console (see below).

To enable and configure OCVPN - CLI:

```
config vpn ocvpn
   set status {enable | disable}
   set poll interval <30 - 120>
   config subnets
      edit 1
```

```
            set subnet 10.1.1.0 255.255.255.0
        next
        edit 2
            set subnet 10.1.2.0 255.255.255.0
        next
    end
end
```

where:

Command	Description
`set status {enable | disable}`	This command enables or disables the service. After a device has been registered with FortiCare, enabling this feature registers the device with the OCVPN cloud service. Disabling causes the device to be unregistered, and removed from the table of VPN members.
`poll interval <30 - 120>`	Set the OCVPN polling interval. Enter an integer value from 30 to 120 (default = 60).
`config subnets`	This is the OCVPN subcommand for configuring the list of participating subnets.

Key exchange

Keys are generated automatically by OCVPN, but without explicit acknowledgment and state management, it would be impossible for the cloud to destroy keys after distribution to customer devices. Permanently storing customer keys in the cloud is undesirable for a host of reasons, so the RegAck request was introduced to effectively address the problem and allow the cloud to destroy keys after they have been installed. One key is generated per customer. When a new member joins, a new key is generated and distributed to all group members at the next poll interval (the default is 60 seconds).

Authentication is handled by SSL and proof of identity is established by the device serial number in the signed RSA certificate. The SN is sent in all messages to the cloud.

If you have a FortiWeb server performing authentication, the process is different. Since the OCVPN microservice doesn't run on the FortiWeb server, OCVPN authentication and secure segregation of customer data is handled as follows:

- FortiWeb extracts the ASN1 CN from the certificate and attaches it to the decrypted HTTP messages forwarded to OCVPN.
- OCVPN checks the presented device SN against the SN included in the certificate ID.
- If they don't match, OCVPN returns '401 Unauthorized' and the authentication transaction is cancelled.

Device polling and controller information

Instead of a central controller actively directing and pushing out the devices in response to network topology changes, FortiOS architecture uses device polling to propagate changes across nodes in the VPN. State changes

are tracked carefully across the system so all devices always have the same view of the network (with some delay in propagating changes due to polling). Similarly the OCVPN cloud always know the state of each device. This is essential to being able to manage the keys properly, and be able to discard them after they have been installed on each device.

The control layer is implemented on each device as a state machine, where information is translated from the member table into a working configuration--with IPsec phase1 and phase2 objects with default parameters, firewall address and address group objects, firewall policies, and static routes. The resulting configuration may be edited normally, e.g. DPD settings, DH group, crypto transform, firewall policy profiles for AV/IPS, etc. This is to provide a level of flexibility and usability.

The control layer's responsibility is to ensure that the network data on any device, and by extension the configuration, always stays in sync with the network view stored in the cloud, and in sync with all the other devices, regardless of intermittent network errors that could occur at any point in the system. The system is designed to handle network errors, changes, and events and keep the IPsec configuration consistently and reliable in sync.

Configuration information is managed in a fixed table: 16 nodes maximum, 16 subnets per node. After the table is populated, full mesh configuration is calculated and installed into the CMDB.

System states

The system is stateless across reboots. It re-registers after reboot, which re-initializes the state of the system. After bootup, the system is stateful across changes and polling interval queries/updates. The state file contains the hostname, current WAN ifname, current WAN IP, assigned slot, current state, previous state, current OCVPN table revision, last OCVPN response code (register/update), last polling response code, number of members, current member bitmask, previous member bitmask. The system uses this state information to track state changes locally and in the cloud.

Possible device states are:

```
enum cvpn_state {
   cvpn_st_none,
   cvpn_st_unregistered,
   cvpn_st_registering,
   cvpn_st_updating,
   cvpn_st_unregistering,
   cvpn_st_acknowledging,
   cvpn_st_registered
};
```

A normal sequence would be registering (updating) -> acknowledging -> registered.

Even though SSL/TCP is stateful and ensures delivery, the OCVPN microservice doesn't run on a FortiWeb SSL termination server. See "Key exchange" on page 112 for more info about how FortiWeb configuration differs. The explicit acknowledgment message (RegAck) ensures the OCVPN service knows when all nodes have received and applied the latest revision of the network information and key.

Debugging and logging

OCVPN debugging and logging is handled through a common API function. All debugs (except polling) are logged to /tmp/ocvpn/log. When the size of the log file exceeds 128k, the file is truncated and only the most recent 32k is saved.

The following diagnose commands may be useful when troubleshooting and debugging OCVPN configurations.

Command	Description
`diag vpn ocvpn`	Top level diagnose command for OCVPN.
`device-state`	Display OCVPN device state.
`log`	Display OCVPN log file from the device.
`status`	Display the current status of the device and last response code from the OCVPN service.
`print-members`	Print the OCVPN member table. This command accesses the OCVPN cloud service to retrieve the latest information, irrespective of the state of the device. It prints the raw JSON responses from OCVPN.

Dynamic DNS configuration

This section describes how to configure a site-to-site VPN, in which one FortiGate unit has a static IP address and the other FortiGate unit has a domain name and a dynamic IP address.

The following topics are included in this section:

Dynamic DNS over VPN concepts

A typical computer has a static IP address and one or more DNS servers to resolve fully qualified domain names (FQDN) into IP addresses. A domain name assigned to this computer is resolved by any DNS server having an entry for the domain name and its static IP address. The IP address never changes or changes only rarely so the DNS server can reliably say it has the correct address for that domain all the time.

Dynamic DNS (DDNS)

It is different when a computer has a dynamic IP address, such as an IP address assigned dynamically by a DHCP server, and a domain name. Computers that want to contact this computer do not know what its current IP address is. To solve this problem there are dynamic DNS (DDNS) servers. These are public servers that store a DNS entry for your computer that includes its current IP address and associated domain name. These entries are kept up to date by your computer sending its current IP address to the DDNS server to ensure its entry is always up to date. When other computers want to contact your domain, their DNS gets your IP address from your DDNS server. To use DDNS servers, you must subscribe to them and usually pay for their services.

When configuring DDNS on your FortiGate unit, go to **Network > DNS** and enable **Enable FortiGuard DDNS**. Then select the interface with the dynamic connection, which DDNS server you have an account with, your domain name, and account information. If your DDNS server is not on the list, there is a generic option where you can provide your DDNS server information.

Routing

When an interface has some form of changing IP address (DDNS, PPPoE, or DHCP assigned address), routing needs special attention. The standard static route cannot handle the changing IP address. The solution is to use the dynamic-gateway command in the CLI. Say for example you already have four static routes, and you have a PPPoE connection over the wan2 interface and you want to use that as your default route.

The route is configured on the dynamic address VPN peer trying to access the static address FortiGate unit.

Configuring dynamic gateway routing - CLI

```
config router static
   edit 5
      set dst 0.0.0.0 0.0.0.0
      set dynamic-gateway enable
      set device wan2
```

```
    next
end
```

For more information on DDNS, see the System Administration handbook chapter.

DDNS over VPN

IPsec VPN expects an IP address for each end of the VPN tunnel. All configuration and communication with that tunnel depends on the IP addresses as reference points. However, when the interface the tunnel is on has DDNS enabled there is no set IP address. The remote end of the VPN tunnel now needs another way to reference your end of the VPN tunnel. This is accomplished using Local ID.

A FortiGate unit that has a domain name and a dynamic IP address can initiate VPN connections anytime. The remote peer can reply to the local FortiGate unit using the source IP address that was sent in the packet header because it is current. Without doing a DNS lookup first, the remote peer runs the risk of the dynamic IP changing before it attempts to connect. To avoid this, the remote peer must perform a DNS lookup for the domain name of to be sure of the dynamic IP address before initiating the connection.

Remote gateway

When configuring the Phase 1 entry for a VPN tunnel, the Remote Gateway determines the addressing method the remote end of the tunnel uses as one of Static IP Address, Dialup User, or Dynamic DNS. There are different fields for each option.

When you select the Dynamic DNS VPN type there is a related field called Dynamic DNS. The Dynamic DNS field is asking for the FQDN of the remote end of the tunnel. It uses this information to look up the IP address of the remote end of the tunnel through the DDNS server associated with that domain name.

Local ID (peer ID)

The Local ID or peer ID can be used to uniquely identify one end of a VPN tunnel. This enables a more secure connection. Also if you have multiple VPN tunnels negotiating, this ensures the proper remote and local ends connect. When you configure it on your end, it is your Local ID. When the remote end connects to you, they see it as your peer ID.

If you are debugging a VPN connection, the Local ID is part of the VPN negotiations. You can use it to help troubleshoot connection problems.

In circumstances where multiple remote dialup VPN tunnels exist, each tunnel must have a peer ID set.

Configuring your Local ID

1. Go to **VPN > IPsec Wizard** and create the new custom tunnel or go to **VPN > IPsec Tunnels** and edit an existing tunnel.
2. Edit the **Phase 1 Proposal** (if it is not available, you may need to click the **Convert To Custom Tunnel** button).
3. In the **Phase 1 Proposal** section, enter your **Local ID**.
4. Select **OK**.

The default configuration is to accept all local IDs (peer IDs). If you have **Local ID** set, the remote end of the tunnel must be configured to accept your local ID.

Accepting a specific Peer ID

1. Go to **VPN > IPsec Tunnels** and create the new custom tunnel or edit an existing tunnel.
2. Edit **Authentication** (if it is not available, you may need to click the **Convert To Custom Tunnel** button).
3. Set **Mode** to **Aggressive**.
4. For **Peer Options**, select **This peer ID**. This option becomes visible only when **Aggressive** mode is selected.
5. In the **Peer ID** field, enter the string the other end of the tunnel used for its local ID.
6. Configure the rest of the Phase 1 entry as required.
7. Select **OK**.

Route-based or policy-based VPN

VPN over dynamic DNS can be configured with either route-based or policy-based VPN settings. Both are valid, but have differences in configuration. Choose the best method based on your requirements. For more information on route-based and policy-based, see IPsec VPN overview on page 27.

Route-based VPN configuration requires two security policies to be configured (one for each direction of traffic) to permit traffic over the VPN virtual interface, and you must also add a static route entry for that VPN interface or the VPN traffic will not reach its destination. See Dynamic DNS configuration on page 115 and Dynamic DNS configuration on page 115.

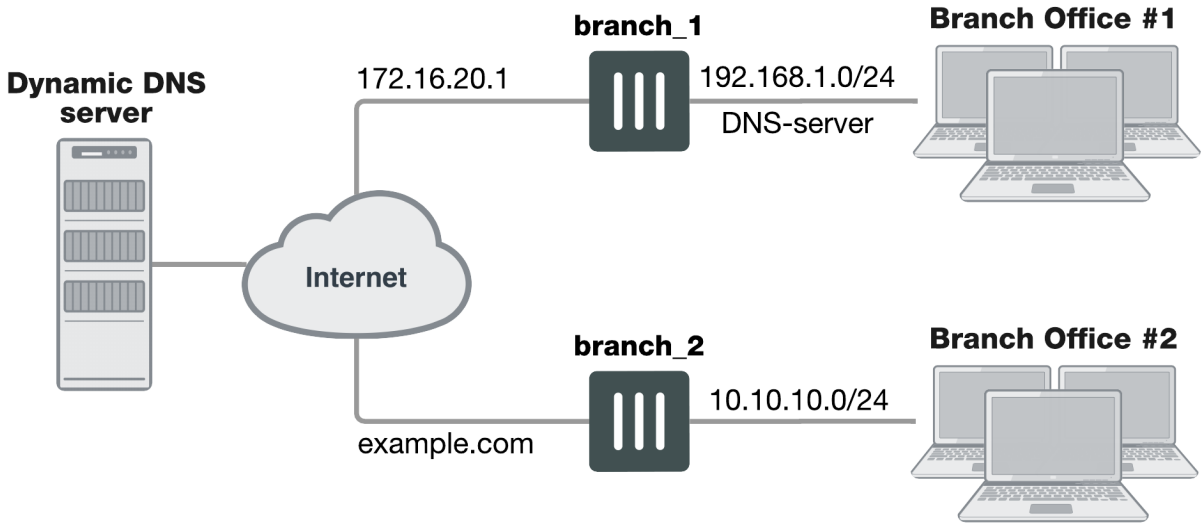
Policy-based VPN configuration uses more complex and often more IPsec security policies, but does not require a static route entry. It has the benefit of being able to configure multiple policies for handling multiple protocols in different ways, such as more scanning of less secure protocols or guaranteeing a minimum bandwidth for protocols such as VoIP. See Dynamic DNS configuration on page 115 and Dynamic DNS configuration on page 115.

DDNS topology

In this scenario, two branch offices each have a FortiGate unit and are connected in a gateway-to-gateway VPN configuration. One FortiGate unit has a domain name (example.com) with a dynamic IP address. See `branch_2` in the figure below.

Whenever the `branch_2` unit connects to the Internet (and possibly also at predefined intervals set by the ISP), the ISP may assign a different IP address to the FortiGate unit. The unit has its domain name registered with a dynamic DNS service. The `branch_2` unit checks in with the DDNS server on a regular basis, and that server provides the DNS information for the domain name, updating the IP address from time to time. Remote peers have to locate the `branch_2` FortiGate unit through a DNS lookup each time to ensure the address they get is current and correct.

Example dynamic DNS configuration



When a remote peer (such as the `branch_1` FortiGate unit above) initiates a connection to `example.com`, the local DNS server looks up and returns the IP address that matches the domain name `example.com`. The remote peer uses the retrieved IP address to establish a VPN connection with the `branch_2` FortiGate unit.

Assumptions

- You have administrator access to both FortiGate units.
- Both FortiGate units have interfaces named wan1 and internal. (If not, you can use the alias feature to assign these labels as “nicknames” to other interfaces to follow this example.)
- Both FortiGate units have the most recent firmware installed, have been configured for their networks, and are currently passing normal network traffic.
- The `branch_2` FortiGate unit has its `wan1` interface defined as a dynamic DNS interface with the domain name of **example.com**.
- A basic gateway-to-gateway configuration is in place (see Gateway-to-gateway configurations on page 1) except one of the FortiGate units has a static domain name and a dynamic IP address instead of a static IP address.
- The FortiGate unit with the domain name is subscribed to one of the supported dynamic DNS services. Contact one of the services to set up an account. For more information and instructions about how to configure the FortiGate unit to push its dynamic IP address to a dynamic DNS server, see the System Administration handbook chapter.

Configuration overview

When a FortiGate unit receives a connection request from a remote VPN peer, it uses IPsec Phase 1 parameters to establish a secure connection and authenticate the VPN peer. Then, if the security policy permits the connection, the FortiGate unit establishes the tunnel using IPsec Phase 2 parameters and applies the security policy. Key management, authentication, and security services are negotiated dynamically through the IKE protocol.

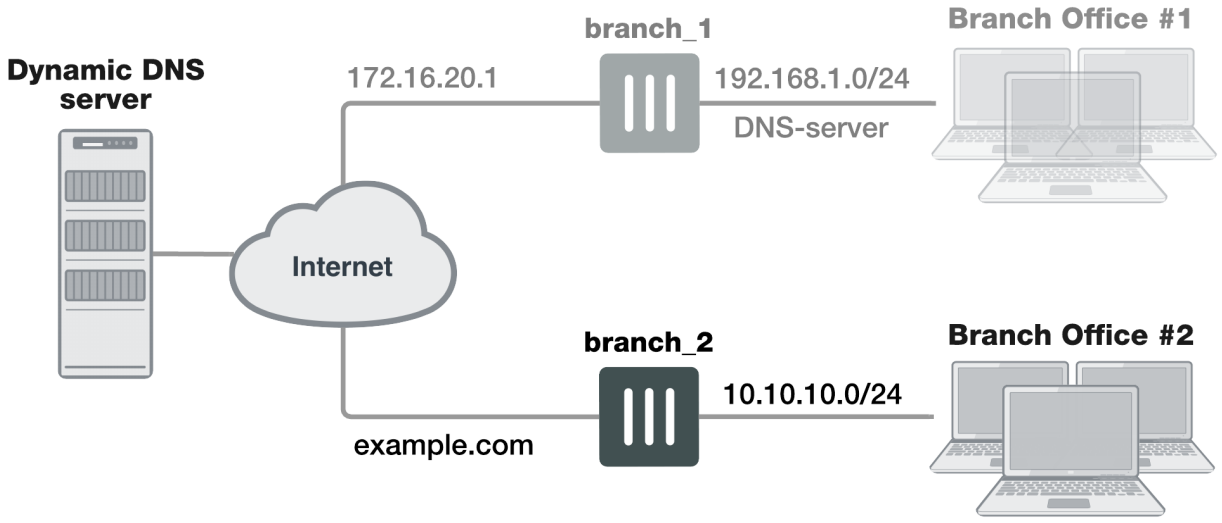
To support these functions, the following general configuration steps must be performed:

- Configure the branch_2 FortiGate unit with the dynamic IP address. This unit uses a Local ID string instead of an IP address to identify itself to the remote peer. See Configuring the dynamically-addressed VPN peer below, which is made up of configuring branch_2's VPN tunnel settings and security policies.
- Configure the fixed-address VPN peer. To initiate a VPN tunnel with the dynamically-addressed peer, this unit must first retrieve the IP address for the domain from the dynamic DNS service. See Configuring the fixed-address VPN peer, which is made up of configuring branch_1's VPN tunnel settings and security policies.

Configuring the dynamically-addressed VPN peer

It is assumed that this FortiGate unit (`branch_2`) has already had its public facing interface, for example the `wan1`, configured with the proper dynamic DNS configuration.

Configuring branch_2, the dynamic address side



Define the Phase 1 parameters needed to establish a secure connection with the remote peer. See Phase 1 parameters on page 46. During this procedure you need to choose if you will be using route-based or policy-based VPNs.

1. Go to **VPN > IPsec Tunnels** and create the new custom tunnel or edit an existing tunnel.
2. Edit **Network** (full configuration options are only available once you click the **Convert To Custom Tunnel** button).
3. Enter the following information:

Remote Gateway	Select **Static IP Address**. The remote peer this FortiGate is connecting to has a static IP public address. If the remote interface is PPPoE do not select **Retrieve default gateway from server**.
IP Address	Enter `172.16.20.1`, the IP address of the public interface to the remote peer.

Interface	Select the Internet-facing interface **wan1** (selected by default).
NAT Traversal	Select **Enable** (selected by default).
Keepalive Frequency	Enter a keepalive frequency (In seconds; set to **10** by default).
Dead Peer Detection	Select a dead peer detection option. **On Idle** will attempt to reestablish VPN tunnels when a connection becomes idle (the idle interval is not a negotiated value). Use of periodic dead peer detection incurs extra overhead. When communicating to large numbers of IKE peers, you should consider using **On Demand**. (set to **On Demand** by default).

4. Edit **Authentication** and complete the following:

Mode	Select **Aggressive**.

5. Edit **Phase 1 Proposal** and complete the following:

Local ID	Enter `example.com`. A character string used by the `branch_2` FortiGate unit to identify itself to the remote peer. This value must be identical to the value in the **This peer ID** field of the Phase 1 remote gateway configuration on the branch_1 remote peer. See Configuration overview on page 118.

6. Open the **Phase 2 Selectors** panel.
   Define the Phase 2 parameters needed to create a VPN tunnel with the remote peer. For details on Phase 2, see Phase 2 parameters on page 66.
7. Enter the following information and select **OK**.

Name	Automatically entered as the name of the VPN tunnel.
Phase 1	Select `branch_2`. The name of the Phase 1 configuration that you defined earlier.

Define security policies to permit communications between the private networks through the VPN tunnel. Route-based and policy-based VPNs require different security policies. For detailed information about creating security policies, see Defining VPN security policies on page 1.

After defining the two address ranges, select one of Creating branch_2 route-ased security policies on page 121 or Creating branch_2 policy-based security policies on page 123 to configure the appropriate VPN policies.

Define VPN connection names for the address ranges of the private networks. These addresses are used in the security policies that permit communication between the networks. For more information, see Defining VPN security policies on page 1.

Define an address name for the IP address and netmask of the private network behind the local FortiGate unit.

1. Go to **Policy & Objects > Addresses**.
2. Select **Create New**.
3. Enter the following information, and select **OK**.

Name	Enter `branch_2_internal`. Enter a meaningful name.
Type	Select **IP/Netmask**.
Subnet / IP Range	Enter `10.10.10.0/24`. Include the netmask or specify a specific range.
Interface	Select **internal**. The interface that will be handling the traffic from the internal network.

Define an address name for the IP address and netmask of the private network behind the remote peer.

4. Select **Create New**.
5. Enter the following information, and select **OK**.

Name	Enter `branch_1_internal`. A meaningful name for the private network at the remote end of the VPN tunnel.
Type	Select **IP/Netmask**.
Subnet / IP Range	Enter `192.168.1.0/24`. Include the netmask. Optionally you can specify a range
Interface	Select **any**. The interface that will be handling the remote VPN traffic on this FortiGate unit. If you are unsure, or multiple interfaces may be handling this traffic use `any`.

Creating branch_2 route-ased security policies

Define ACCEPT security policies to permit communication between the branch_2 and branch_1 private networks. Once the route-based policy is configured a routing entry must be configured to route traffic over the VPN interface.

Define a policy to permit the branch_2 local FortiGate unit to initiate a VPN session with the branch_1 VPN peer.

1. Go to **Policy & Objects > IPv4 Policy** and select **Create New**.
2. Enter the following information, and select **OK**.

Name	Enter an appropriate name for the policy.

Incoming Interface	Select **internal**. The interface that connects to the private network behind this FortiGate unit.
Outgoing Interface	Select **branch_2**. The VPN Tunnel (IPsec Interface).
Source	Select **branch_2_internal**. Select the address name for the private network behind this FortiGate unit.
Destination Address	Select **branch_1_internal**. The address name the private network behind the remote peer.
Action	Select **ACCEPT**.
NAT	Disable **NAT**.
Comments	Route-based: Initiate a branch_2 to branch_1 VPN tunnel.

Define a policy to permit the branch_1 remote VPN peer to initiate VPN sessions.

3. Select **Create New**.
4. Enter the following information, and select **OK**.

Name	Enter an appropriate name for the policy.
Incoming Interface	Select **branch_2**. The VPN Tunnel (IPsec Interface).
Outgoing Interface	Select **internal**. The interface connecting the private network behind this FortiGate unit.
Source	Select **branch_1_internal**. The address name for the private network behind the remote peer.
Destination Address	Select **branch_2_internal**. The address name for the private network behind this FortiGate unit.
Action	Select **ACCEPT**.
NAT	Disable **NAT**.
Comments	Route-based: Initiate a branch_1 to branch_2 internal VPN tunnel.

5. Optionally configure any other security policy settings you require such as UTM or traffic shaping for this policy.
6. Place these policies in the policy list above any other policies having similar source and destination addresses. This will ensure VPN traffic is matched against the VPN policies before any other policies.

Creating routing entry for VPN interface - CLI

```
config router static
   edit 5
      set dst 0.0.0.0 0.0.0.0
```

```
        set dynamic-dateway enable
        set device wan1
    next
end
```

This routing entry must be added in the CLI because the dynamic-gateway option is not available in the web-based manager.

Creating branch_2 policy-based security policies

Define an IPsec policy to permit VPN sessions between the private networks. Define an IPsec policy to permit the VPN sessions between the local branch_2 unit and the remote branch_1 unit.

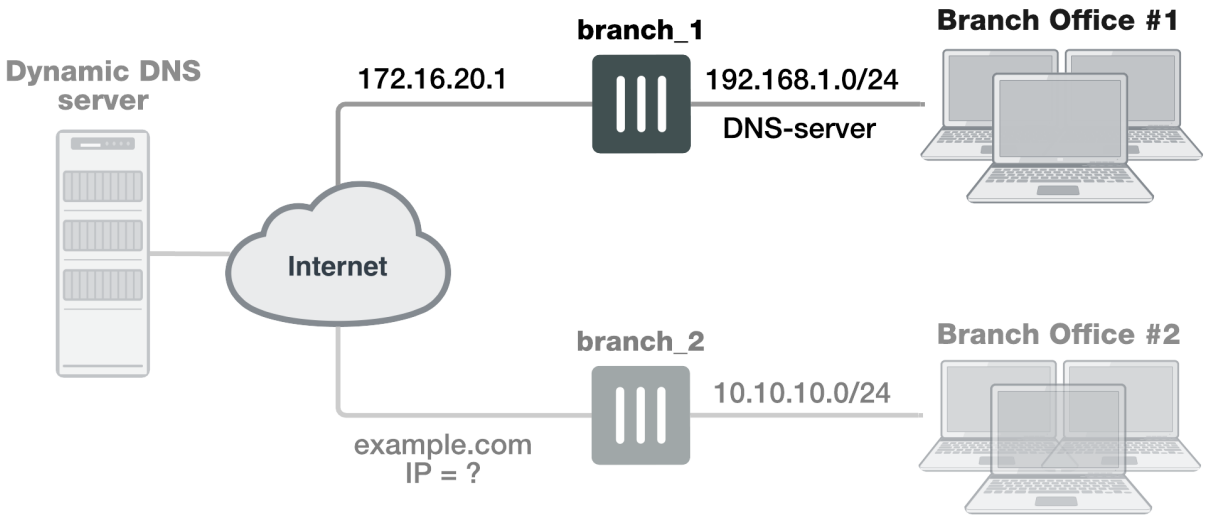
1. Go to **Policy & Objects > IPv4 Policy** and select **Create New**.
2. Enter the following information, and select **OK**.

Name	Enter an appropriate name for the policy.
Incoming Interface	Select **internal**. The interface connecting the private network behind this FortiGate unit.
Outgoing Interface	Select **wan1**. The FortiGate unit's public interface.
Source	Select **branch_2_internal**. The address name for the private network behind this local FortiGate unit.
Destination Address	Select **branch_1_internal**. The address name for the private network behind branch_1, the remote peer.
Action	Select **IPsec**. Under **VPN Tunnel**, select **branch_2** from the drop-down list. The name of the Phase 1 tunnel. Select **Allow traffic to be initiated from the remote site**.
Comments	Policy-based: allows traffic in either direction to initiate the VPN tunnel.

3. Optionally configure any other security policy settings you require such as UTM or traffic shaping for this policy.
4. Place these policies in the policy list above any other policies having similar source and destination addresses. This will ensure VPN traffic is matched against the VPN policies before any other policies.

Configuring the fixed-address VPN peer

The fixed-address VPN peer, branch_1, needs to retrieve the IP address from the dynamic DNS service to initiate communication with the dynamically-addressed peer, branch_2. It also depends on the peer ID (local ID) to initiate the VPN tunnel with branch_2.



Define the Phase 1 parameters needed to establish a secure connection with the remote peer. For more information, see Phase 1 parameters on page 46.

1. Go to **VPN > IPsec Tunnels** and create the new custom tunnel or edit an existing tunnel.
2. Edit **Network** (if it is not available, you may need to click the **Convert to Custom Tunnel** button).
3. Enter the following information and select **OK**.

Remote Gateway	Select **Dynamic DNS**. The remote peer this FortiGate is connecting to has a dynamic IP address.
Dynamic DNS	Type the fully qualified domain name of the remote peer (for example, `example.com`).
Interface	Select **wan1**. The public facing interface on the fixed-address FortiGate unit.
Mode Config	Select **Aggressive**.
Peer Options	Select **This peer ID**, and enter `example.com`. This option only appears when the mode is set to Aggressive. The identifier of the FortiGate unit with the dynamic address.

4. Edit **Authentication**, enter the following information and select **OK**.

Peer Options	Select **This peer ID**, and enter `example.com`. This option only appears when the authentication method is set to **Signature**. The identifier of the FortiGate unit with the dynamic address.

5. Define the Phase 2 parameters needed to create a VPN tunnel with the remote peer. See Phase 2 parameters on page 66. Enter these settings in particular:

Name	Enter `branch_1_p2`. A name to identify this Phase 2 configuration.

Phase 1	Select **branch_1**. The name of the Phase 1 configuration that you defined for the remote peer. You can select the name of the remote gateway from the Dynamic DNS part of the list.

The `branch_1` FortiGate unit has a fixed IP address and will be connecting to the `branch_2` FortiGate unit that has a dynamic IP address and a domain name of `example.com`. Remember if you are using route-based security policies that you must add a route for the VPN traffic.

Defining address ranges for branch_1 security policies

As with `branch_2` previously, `branch_1` needs address ranges defined as well. See Defining policy addresses on page 1.

1. Go to **Policy & Objects > Addresses** and select **Create New > Address**.
2. Enter the following information, and select **OK**.

Name	Enter `branch_2_internal`. A meaningful name for the private network behind the `branch_2` FortiGate unit.
Type	Select **IP/Netmask**.
Subnet / IP Range	Enter `10.10.10.0/24`. Include the netmask or specify a specific range.
Interface	Select **internal**. This is the interface on this FortiGate unit that will be handling with this traffic.

3. Define an address name for the IP address and netmask of the private network behind the remote peer.
4. Create another address. Enter the following information, and select **OK**.

Name	Enter `branch_1_internal`. A meaningful name for the private network behind the `branch_1` peer.
Type	Select **IP/Netmask**.
Subnet / IP Range	Enter `192.168.1.0/24`. Include the netmask or specify a specific range.
Interface	Select **any**. The interface on this FortiGate unit that will be handling with this traffic. If you are unsure, or multiple interfaces may be handling this traffic use `any`.

Creating branch_1 route-based security policies

Define an ACCEPT security policy to permit communications between the source and destination addresses. See Defining VPN security policies on page 1.

1. Go to **Policy & Objects > IPv4 Policy** and select **Create New**.
2. Enter the following information, and select **OK**.

Name	Enter an appropriate name for the policy.
Incoming Interface	Select **internal**. The interface that connects to the private network behind the `branch_1` FortiGate unit.
Outgoing Interface	Select **branch_1**. The VPN Tunnel (IPsec Interface) you configured earlier.
Source	Select **branch_1_internal**. The address name that you defined for the private network behind this FortiGate unit.
Destination Address	Select **branch_2_internal**. The address name that you defined for the private network behind the `branch_2` peer.
Action	Select **ACCEPT**.
NAT	Disable **NAT**.
Comments	Internal -> branch2

To permit the remote client to initiate communication, you need to define a security policy for communication in that direction.

3. Select **Create New**.
4. Enter the following information, and select **OK**.

Name	Enter an appropriate name for the policy.
Incoming Interface	Select **branch_1**. The VPN Tunnel (IPsec Interface) you configured earlier.
Outgoing Interface	Select **internal**. The interface that connects to the private network behind this FortiGate unit.
Source	Select **branch_2_internal**. The address name that you defined for the private network behind the `branch_2` remote peer.
Destination Address	Select **branch_1_internal**. The address name that you defined for the private network behind this FortiGate unit.
Action	Select **ACCEPT**.
NAT	Disable **NAT**.
Comments	branch_2 -> Internal

Creating branch_1 policy-based security policies

A policy-based security policy allows you the flexibility to allow inbound or outbound traffic or both through this single policy.

This policy-based IPsec VPN security policy allows both inbound and outbound traffic

1. Go to **Policy & Objects > IPv4 Policy** and select **Create New**.
2. Enter the following information, and select **OK**.

Incoming Interface	Select **internal**. The interface that connects to the private network behind this FortiGate unit.
Outgoing Interface	Select **wan1**. The FortiGate unit's public interface.
Source	Select **branch_1_internal**. The address name that you defined for the private network behind this FortiGate unit.
Destination Address	Select **branch_2_internal**. The address name that you defined for the private network behind the remote peer.
Action	Select **IPsec**. Under **VPN Tunnel**, select **branch_1** from the drop-down list. The name of the Phase 1 tunnel. Select **Allow traffic to be initiated from the remote site**.

3. Place this security policy in the policy list above any other policies having similar source and destination addresses.

Results

Once both ends are configured, you can test the VPN tunnel.

To test the VPN initiated by branch_2

1. On branch_2, go to **Monitor > IPsec Monitor**.
   All IPsec VPN tunnels will be listed on this page, no matter if they are connected or disconnected.
2. Select the tunnel listed for branch_2, and select the status column for that entry.
   The status will say **Bring Up** and remote port, incoming and outgoing data will all be zero. This indicates an inactive tunnel. When you right-click and select **Bring Up**, the FortiGate will try to set up a VPN session over this tunnel. If it is successful, Bring Up will change to Active, and the arrow icon will change to a green up arrow icon.
3. If this does not create a VPN tunnel with increasing values for incoming and outgoing data, you need to start troubleshooting:

To test the VPN initiated by branch_1

1. On branch_1, go to **Monitor > IPsec Monitor**.
2. Select the tunnel listed for branch_1, and select the status column.
   The difference between branch_2 and branch_1 at this point is that the tunnel entry for branch-1 will not have a remote gateway IP address. It will be resolved when the VPN tunnel is started.
3. If this does not create a VPN tunnel with increasing values for incoming and outgoing data, you need to start troubleshooting.

Some troubleshooting ideas include:

- If there was no entry for the tunnel on the monitor page, check the Auto Key (IKE) page to verify the Phase 1 and Phase 2 entries exist.
- Check the security policy or policies, and ensure there is an outgoing policy as a minimum.
- Check that you entered a local ID in the Phase 1 configuration, and that branch_1 has the same local ID.
- Ensure the local DNS server has an up-to-date DNS entry for exmaple.com.

For more information, see Troubleshooting on page 1.

FortiClient dialup-client configuration

The FortiClient Endpoint Security application is an IPsec VPN client with antivirus, antispam and firewall capabilities. This section explains how to configure dialup VPN connections between a FortiGate unit and one or more FortiClient Endpoint Security applications.

FortiClient users are usually mobile or remote users who need to connect to a private network behind a FortiGate unit. For example, the users might be employees who connect to the office network while traveling or from their homes.

For greatest ease of use, the FortiClient application can download the VPN settings from the FortiGate unit to configure itself automatically.

The following topics are included in this section:

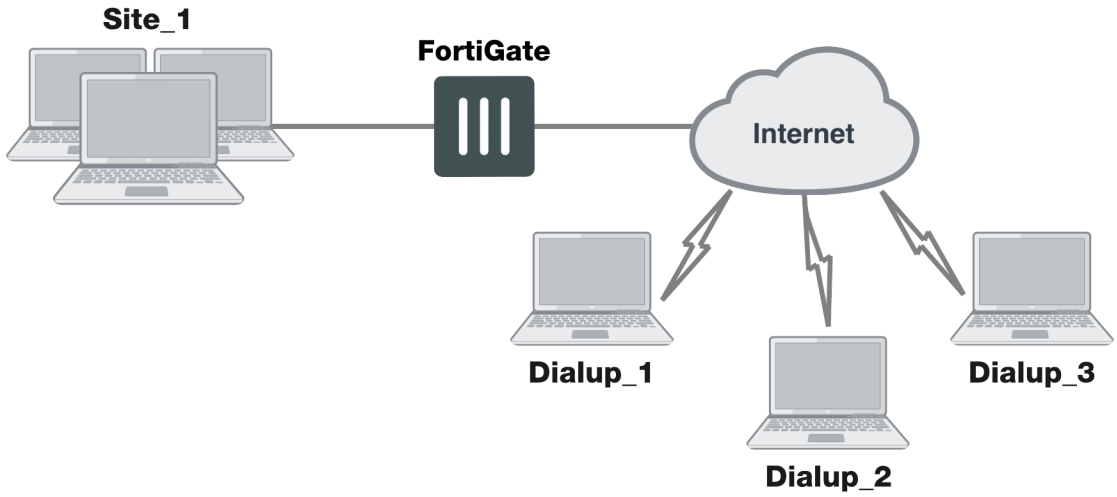
Configuration overview

Dialup users typically obtain dynamic IP addresses from an ISP through Dynamic Host Configuration Protocol (DHCP) or Point-to-Point Protocol over Ethernet (PPPoE). Then, the FortiClient Endpoint Security application initiates a connection to a FortiGate dialup server.

By default the FortiClient dialup client has the same IP address as the host PC on which it runs. If the host connects directly to the Internet, this is a public IP address. If the host is behind a NAT device, such as a router, the IP address is a private IP address. The NAT device must be NAT traversal (NAT-T) compatible to pass encrypted packets (see Phase 1 parameters on page 46). The FortiClient application also can be configured to use a virtual IP address (VIP). For the duration of the connection, the FortiClient application and the FortiGate unit both use the VIP address as the IP address of the FortiClient dialup client.

The FortiClient application sends its encrypted packets to the VPN remote gateway, which is usually the public interface of the FortiGate unit. It also uses this interface to download VPN settings from the FortiGate unit. See Automatic configuration of FortiClient dialup clients on page 129.

Example FortiClient dialup-client configuration



Peer identification

The FortiClient application can establish an IPsec tunnel with a FortiGate unit configured to act as a dialup server. When the FortiGate unit acts as a dialup server, it does not identify the client using the Phase 1 remote gateway address. The IPsec tunnel is established if authentication is successful and the IPsec security policy associated with the tunnel permits access. If configured, the FortiGate unit could also require FortiClient registration, that is, the remote user would be required to have FortiClient installed before connection is completed.

Automatic configuration of FortiClient dialup clients

The FortiClient application can obtain its VPN settings from the FortiGate VPN server. FortiClient users need to know only the FortiGate VPN server IP address and their username and password on the FortiGate unit.

The FortiGate unit listens for VPN policy requests from clients on TCP port 8900. When the dialup client connects:

- The client initiates a Secure Sockets Layer (SSL) connection to the FortiGate unit.
- The FortiGate unit requests a user name and password from the FortiClient user. Using these credentials, it authenticates the client and determines which VPN policy applies to the client.
- Provided that authentication is successful, the FortiGate unit downloads a VPN policy to the client over the SSL connection. The information includes IPsec Phase 1 and Phase 2 settings, and the IP addresses of the private networks that the client is authorized to access.
- The client uses the VPN policy settings to establish an IPsec Phase 1 connection and Phase 2 tunnel with the FortiGate unit.

FortiClient-to-FortiGate VPN configuration steps

Configuring dialup client capability for FortiClient dialup clients involves the following general configuration steps:

1. If you will be using VIP addresses to identify dialup clients, determine which VIP addresses to use. As a precaution, consider using VIP addresses that are not commonly used.

2. Configure the FortiGate unit to act as a dialup server. See Configure the FortiGate unit on page 1.
3. If the dialup clients will be configured to obtain VIP addresses through DHCP over IPsec, configure the FortiGate unit to act as a DHCP server or to relay DHCP requests to an external DHCP server.
4. Configure the dialup clients. See Configure the FortiClient Endpoint Security application on page 1.

Using virtual IP addresses

When the FortiClient host PC is located behind a NAT device, unintended IP address overlap issues may arise between the private networks at the two ends of the tunnel. For example, the client's host might receive a private IP address from a DHCP server on its network that by co-incidence is the same as a private IP address on the network behind the FortiGate unit. A conflict will occur in the host's routing table and the FortiClient Endpoint Security application will be unable to send traffic through the tunnel. Configuring virtual IP (VIP) addresses for FortiClient applications prevents this problem.

Using VIPs ensures that client IP addresses are in a predictable range. You can then define security policies that allow access only to that source address range. If you do not use VIPs, the security policies must allow all source addresses because you cannot predict the IP address for a remote mobile user.

The FortiClient application must not have the same IP address as any host on the private network behind the FortiGate unit or any other connected FortiClient application. You can ensure this by reserving a range of IP addresses on the private network for FortiClient users. Or, you can assign FortiClient VIPs from an uncommonly used subnet such as 10.254.254.0/24 or 192.168.254.0/24.

You can reserve a VIP address for a particular client according to its device MAC address and type of connection. The DHCP server then always assigns the reserved VIP address to the client. For more information about this feature, see the "dhcp reserved-address" section in the "system" chapter of the FortiGate CLI Reference.

On the host computer, you can find out the VIP address that the FortiClient Endpoint Security application is using. For example, in Windows command prompt, type `ipconfig /all`

On Linux or Mac OS X, type `ifconfig` in a terminal window. The output will also show the IP address that has been assigned to the host Network Interface Card (NIC).

It is best to assign VIPs using DHCP over IPsec. The FortiGate dialup server can act as a DHCP server or relay requests to an external DHCP server. You can also configure VIPs manually on FortiClient applications, but it is more difficult to ensure that all clients use unique addresses.

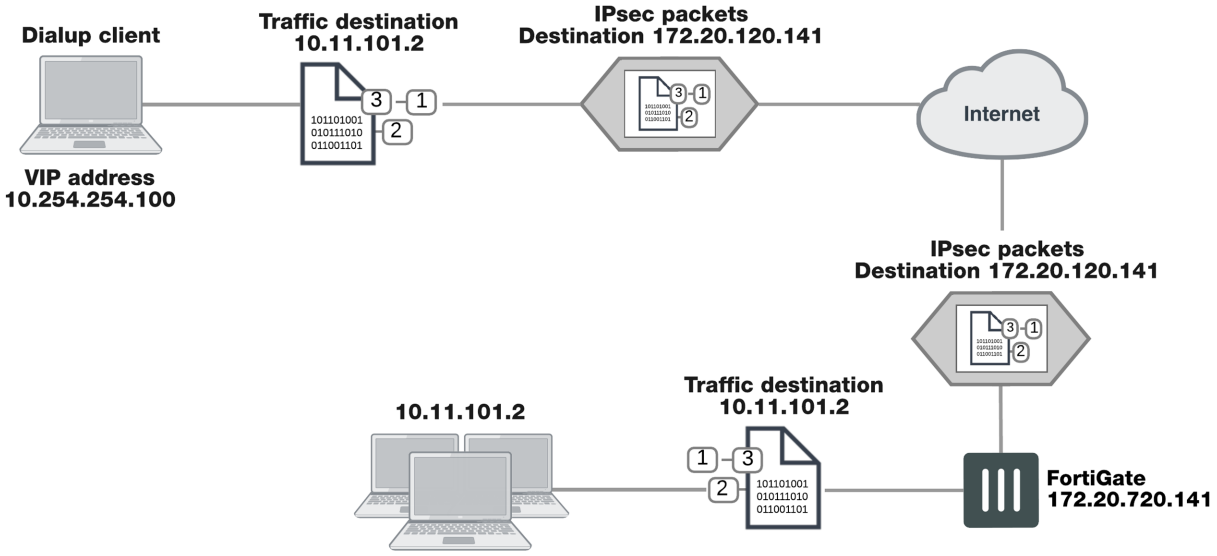
If you assign a VIP on the private network behind the FortiGate unit and enable DHCP-IPsec (a Phase 2 advanced option), the FortiGate unit acts as a proxy on the local private network for the FortiClient dialup client. Whenever a host on the network behind the dialup server issues an ARP request for the device MAC address of the FortiClient host, the FortiGate unit answers the ARP request on behalf of the FortiClient host and forwards the associated traffic to the FortiClient host through the tunnel. For more information, see Phase 2 parameters on page 66.

FortiGate units fully support RFC 3456. The FortiGate DHCP over IPsec feature can be enabled to allocate VIP addresses to FortiClient dialup clients using a FortiGate DHCP server.

The figure below shows an example of a FortiClient-to-FortiGate VPN where the FortiClient application is assigned a VIP on an uncommonly used subnet. The diagram also shows that while the destination for the

information in the encrypted packets is the private network behind the FortiGate unit, the destination of the IPsec packets themselves is the public interface of the FortiGate unit that acts as the end of the VPN tunnel.

IP address assignments in a FortiClient dialup-client configuration



Assigning VIPs by RADIUS user group

If you use XAuth authentication, you can assign users the virtual IP address stored in the Framed-IP-Address field of their record on the RADIUS server. (See RFC 2865 and RFC 2866 for more information about RADIUS fields.) To do this:

- Set the DHCP server **IP Assignment Mode** to **User-group defined method**. This is an Advanced setting. See Configuring a DHCP server on a FortiGate interface on page 135.
- Create a new firewall user group and add the RADIUS server to it.
- In your Phase 1 settings, configure the FortiGate unit as an XAuth server and select from **User Group** the new user group that you created. For more information, see Phase 1 parameters on page 46.
- Configure the FortiClient application to use XAuth. See Configuration overview on page 128.

FortiClient dialup-client infrastructure requirements

- To support policy-based VPNs, the FortiGate dialup server may operate in either NAT mode or transparent mode. NAT mode is required if you want to create a route-based VPN.
- If the FortiClient dialup clients will be configured to obtain VIP addresses through FortiGate DHCP relay, a DHCP server must be available on the network behind the FortiGate unit and the DHCP server must have a direct route to the FortiGate unit.
- If the FortiGate interface to the private network is not the default gateway, the private network behind the FortiGate unit must be configured to route IP traffic destined for dialup clients back (through an appropriate gateway) to the FortiGate interface to the private network. As an alternative, you can configure the IPsec security policy on the FortiGate unit to perform inbound NAT on IP packets. Inbound NAT translates the source addresses of inbound decrypted packets into the IP address of the FortiGate interface to the local private network.

Configuring the FortiGate unit

Configuring the FortiGate unit to establish VPN connections with FortiClient Endpoint Security users involves the following steps:

- Configure the VPN settings
- If the dialup clients use automatic configuration, configure the FortiGate unit as a VPN policy server
- If the dialup clients obtain VIP addresses by DHCP over IPsec, configure an IPsec DHCP server or relay

The procedures in this section cover basic setup of policy-based and route-based VPNs compatible with FortiClient Endpoint Security. A route-based VPN is simpler to configure.

The IPsec VPN Wizard greatly simplifies IPsec VPN tunnel creation for route-based tunnels.

To configure FortiGate unit VPN settings to support FortiClient users, you need to:

- Configure the FortiGate Phase 1 VPN settings
- Configure the FortiGate Phase 2 VPN settings
- Add the security policy

On the local FortiGate unit, define the Phase 1 configuration needed to establish a secure connection with the FortiClient peer. See Phase 1 parameters on page 46.

1. Go to **VPN > IPsec Tunnels** and create the new custom tunnel or edit an existing tunnel.
2. Edit **Network** (full configuration options are only available once you click the **Convert To Custom Tunnel** button).
3. Enter these settings in particular:

Remote Gateway	Select **Dialup User**.
IP Address	Enter the IP address of the remote peer.
Interface	Select the interface through which clients connect to the FortiGate unit.
Mode Config	When enabled, further options become available: • **Client Address Range** • **Subnet Mask** • **Use System DNS** • **DNS Server** • **Enable IPv4 Split Tunnel**
Authentication Method	Select **Pre-shared Key**.
Pre-shared Key	Enter the pre-shared key. This must be the same preshared key provided to the FortiClient users.
Peer option	Select **Any peer ID**.

4. Edit **Authentication** and enter the following information:

Method	Select **Pre-shared Key**.
Pre-shared Key	Enter the pre-shared key. This must be the same preshared key provided to the FortiClient users.
Peer Options	Set **Accept Types** to **Any peer ID**.

5. Define the Phase 2 parameters needed to create a VPN tunnel with the FortiClient peer. See Phase 2 parameters on page 66. Enter these settings in particular:

Name	Enter a name to identify this Phase 2 configuration.
Phase 1	Select the name of the Phase 1 configuration that you defined.
Advanced	Select to configure the following optional setting.
DHCP-IPsec	Select if you provide virtual IP addresses to clients using DHCP.

6. Define names for the addresses or address ranges of the private networks that the VPN links. These addresses are used in the security policies that permit communication between the networks. For more information, see Defining policy addresses on page 1.

   Enter these settings in particular:
   - Define an address name for the individual address or the subnet address that the dialup users access through the VPN.
   - If FortiClient users are assigned VIP addresses, define an address name for the subnet to which these VIPs belong.

4. Define security policies to permit communication between the private networks through the VPN tunnel. Route-based and policy-based VPNs require different security policies. For detailed information about creating security policies, see Defining VPN security policies on page 1.

If the security policy, which grants the VPN Connection is limited to certain services, DHCP must be included, otherwise the client won't be able to retrieve a lease from the FortiGate's (IPsec) DHCP server, because the DHCP Request (coming out of the tunnel) will be blocked.

Route-based VPN security policies

Define an ACCEPT security policy to permit communications between the source and destination addresses.

1. Go to **Policy & Objects > IPv4 Policy** and select **Create New**.
2. Enter these settings in particular:

Name	Enter an appropriate name for the policy.
Incoming Interface	Select the VPN Tunnel (IPsec Interface) you configured in Step "Configuration overview" on page 128.
Outgoing Interface	Select the interface that connects to the private network behind this FortiGate unit.

Source	Select **all**.
Destination Address	Select **all**.
Action	Select **ACCEPT**.
NAT	Disable **NAT**.

If you want to allow hosts on the private network to initiate communications with the FortiClient users after the tunnel is established, you need to define a security policy for communication in that direction.

1. Go to **Policy & Objects > IPv4 Policy** and select **Create New**.
2. Enter these settings in particular:

Incoming Interface	Select the interface that connects to the private network behind this FortiGate unit.
Outgoing Interface	Select the interface that connects to the private network behind this FortiGate unit.
Source	Select **all**.
Destination Address	Select **all**.
Action	Select **ACCEPT**.
NAT	Disable **NAT**.

Policy-based VPN security policy

Define an IPsec security policy to permit communications between the source and destination addresses.

1. Go to **Policy & Objects > IPv4 Policy** and select **Create New**.
2. Enter these settings in particular:

Incoming Interface	Select the interface that connects to the private network behind this FortiGate unit.
Outgoing Interface	Select the FortiGate unit's public interface.
Source	Select the address name that you defined in Step "Configuration overview" on page 128 for the private network behind this FortiGate unit.
Destination Address	If FortiClient users are assigned VIPs, select the address name that you defined for the VIP subnet. Otherwise, select **all**.
Action	Select **IPsec**. Under **VPN Tunnel**, select the name of the Phase 1 configuration that you created in Step "Configuration overview" on page 128 from the drop-down list. Select **Allow traffic to be initiated from the remote site**.

Place VPN policies in the policy list above any other policies having similar source and destination addresses.

Configuring the FortiGate unit as a VPN policy server

When a FortiClient application set to automatic configuration connects to the FortiGate unit, the FortiGate unit requests a user name and password. If the user supplies valid credentials, the FortiGate unit downloads the VPN settings to the FortiClient application.

You must do the following to configure the FortiGate unit to work as a VPN policy server for FortiClient automatic configuration:

1. Create user accounts for FortiClient users.
2. Create a user group for FortiClient users and the user accounts that you created in step 1.
3. Connect to the FortiGate unit CLI and configure VPN policy distribution as follows:

```
config vpn ipsec forticlient
   edit <policy_name>
      set phase2name <tunnel_name>
      set usergroupname <group_name>
      set status enable
   end
```

`<tunnel_name>` must be the Name you specified in the step 2 of Configuration overview on page 128.
`<group_name>` must be the name of the user group your created for FortiClient users.

Configuring DHCP services on a FortiGate interface

If the FortiClient dialup clients are configured to obtain a VIP address using DHCP, configure the FortiGate dialup server to either:

- Relay DHCP requests to a DHCP server behind the FortiGate unit (see Configuring DHCP relay on a FortiGate interface on page 135 below).
- Act as a DHCP server (see Configuring a DHCP server on a FortiGate interface on page 135).

Note that DHCP services are typically configured during the interface creation stage, but you can return to an interface to modify DHCP settings if need be.

Configuring DHCP relay on a FortiGate interface

1. Go to **Network > Interfaces** and select the interface that you want to relay DHCP.
2. Enable **DHCP Server**, and create a new DHCP **Address Range** and **Netmask**.
3. Open the **Advanced...** menu and set **Mode** to **Relay**.
4. Enter the **DHCP Server IP**.
5. Select **OK**.

Configuring a DHCP server on a FortiGate interface

1. Go to **Network > Interfaces** and select the interface that you want to act as a DHCP server.
2. Enable **DHCP Server**, and create a new DHCP **Address Range** and **Netmask**.
3. Set **Default Gateway** to **Specify**, and enter the IP address of the default gateway that the DHCP server assigns to DHCP clients.
4. Set **DNS Server** to **Same as System DNS**. If you want to use a different DNS server for VPN clients, select **Specify** and enter an IP address in the available field.

5. Open the **Advanced...** menu and set **Mode** to **Server**.
6. Select **OK**.

Configure the FortiClient Endpoint Security application

The following procedure explains how to configure the FortiClient Endpoint Security application to communicate with a remote FortiGate dialup server using the VIP address that you specify manually. These procedures are based on FortiClient 5.4.1.

Configuring FortiClient

This procedure explains how to configure the FortiClient application manually using the default IKE and IPsec settings. For more information, refer to the FortiClient Administration Guide.

1. Go to **Remote Access** and select the **Settings** icon.
2. Select **Add a new connection**, set the new VPN connection to **IPsec VPN**, and complete following information:

Connection Name	Enter a descriptive name for the connection.
Remote Gateway	Enter the IP address or the fully qualified domain name (FQDN) of the remote gateway.
Authentication Method	Select **Pre-shared Key** and enter the pre-shared key in the field provided.
Authentication (XAuth)	Extended Authentication (XAuth) increases security by requiring additional user authentication in a separate exchange at the end of the VPN Phase 1 negotiation. The FortiGate unit challenges the user for a user name and password. It then forwards the user's credentials to an external RADIUS or LDAP server for verification. Implementation of XAuth requires configuration at both the FortiGate unit and the FortiClient application.

3. Select **OK**.

Adding XAuth authentication

For information about configuring a FortiGate unit as an XAuth server, see Phase 1 parameters on page 46. The following procedure explains how to configure the FortiClient application.

Note that XAuth is not compatible with IKE version 2.

For more information on configuring XAuth authentication, see the FortiClient Administration Guide.

FortiGate dialup-client configurations

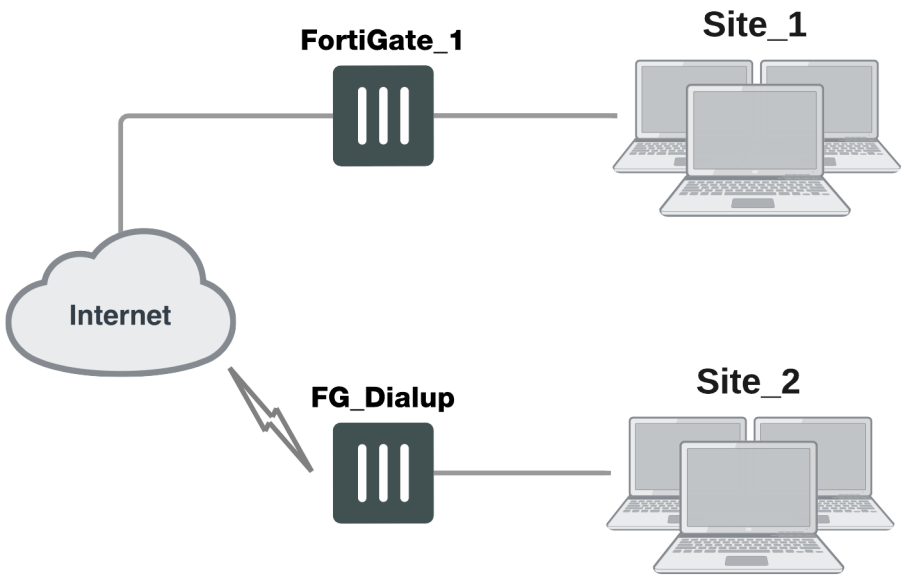
This section explains how to set up a FortiGate dialup-client IPsec VPN. In a FortiGate dialup-client configuration, a FortiGate unit with a static IP address acts as a dialup server and a FortiGate unit having a dynamic IP address initiates a VPN tunnel with the FortiGate dialup server.

The following topics are included in this section:

Configuration overview

A dialup client can be a FortiGate unit. The FortiGate dialup client typically obtains a dynamic IP address from an ISP through the Dynamic Host Configuration Protocol (DHCP) or Point-to-Point Protocol over Ethernet (PPPoE) before initiating a connection to a FortiGate dialup server.

Example FortiGate dialup-client configuration



In a dialup-client configuration, the FortiGate dialup server does not rely on a Phase 1 remote gateway address to establish an IPsec VPN connection with dialup clients. As long as authentication is successful and the IPsec security policy associated with the tunnel permits access, the tunnel is established.

Several different ways to authenticate dialup clients and restrict access to private networks based on client credentials are available. To authenticate FortiGate dialup clients and help to distinguish them from FortiClient dialup clients when multiple clients will be connecting to the VPN through the same tunnel, best practices dictate that you assign a unique identifier (local ID or peer ID) to each FortiGate dialup client. For more information, see Phase 1 parameters on page 46.

Whenever you add a unique identifier (local ID) to a FortiGate dialup client for identification purposes, you must select Aggressive mode on the FortiGate dialup server and also specify the identifier as a peer ID on the FortiGate dialup server. For more information, see Phase 1 parameters on page 46.

Users behind the FortiGate dialup server cannot initiate the tunnel because the FortiGate dialup client does not have a static IP address. After the tunnel is initiated by users behind the FortiGate dialup client, traffic from the private network behind the FortiGate dialup server can be sent to the private network behind the FortiGate dialup client.

Encrypted packets from the FortiGate dialup client are addressed to the public interface of the dialup server. Encrypted packets from the dialup server are addressed either to the public IP address of the FortiGate dialup client (if the dialup client connects to the Internet directly), or if the FortiGate dialup client is behind a NAT device, encrypted packets from the dialup server are addressed to the public IP address of the NAT device.

If a router with NAT capabilities is in front of the FortiGate dialup client, the router must be NAT-T compatible for encrypted traffic to pass through the NAT device. For more information, see Phase 1 parameters on page 46.

When the FortiGate dialup server decrypts a packet from the FortiGate dialup client, the source address in the IP header may be one of the following values, depending on the configuration of the network at the far end of the tunnel:

- If the FortiGate dialup client connects to the Internet directly, the source address will be the private IP address of a host or server on the network behind the FortiGate dialup client.
- If the FortiGate dialup client is behind a NAT device, the source address will be the public IP address of the NAT device.

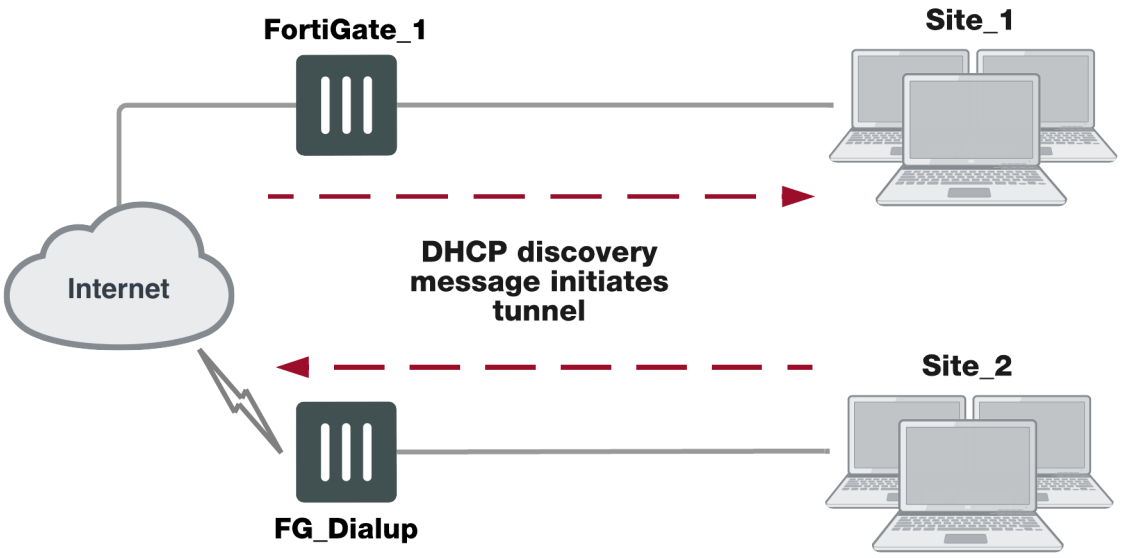
In some cases, computers on the private network behind the FortiGate dialup client may (by co-incidence) have IP addresses that are already used by computers on the network behind the FortiGate dialup server. In this type of situation (ambiguous routing), conflicts may occur in one or both of the FortiGate routing tables and traffic destined for the remote network through the tunnel may not be sent.

In many cases, computers on the private network behind the FortiGate dialup client will most likely obtain IP addresses from a local DHCP server behind the FortiGate dialup client. However, unless the local and remote networks use different private network address spaces, unintended ambiguous routing and IP-address overlap issues may arise.

To avoid these issues, you can configure FortiGate DHCP relay on the dialup client instead of using a DHCP server on the network behind the dialup client. The FortiGate dialup client can be configured to relay DHCP requests from the local private network to a DHCP server that resides on the network behind the FortiGate dialup server. You configure the FortiGate dialup client to pass traffic from the local private network to the remote network by enabling FortiGate DHCP relay on the FortiGate dialup client interface that is connected to the local private network.

Afterward, when a computer on the network behind the dialup client broadcasts a DHCP request, the dialup client relays the message through the tunnel to the remote DHCP server. The remote DHCP server responds with a private IP address for the computer. To avoid ambiguous routing and network overlap issues, the IP addresses assigned to computers behind the dialup client cannot match the network address space used by the private network behind the FortiGate dialup server.

Preventing network overlap in a FortiGate dialup-client configuration



When the DHCP server resides on the private network behind the FortiGate dialup server, the IP destination address specified in the IPsec security policy on the FortiGate dialup client must refer to that network.

You must add a static route to the DHCP server FortiGate unit if it is not directly connected to the private network behind the FortiGate dialup server; its IP address does not match the IP address of the private network. Also, the destination address in the IPsec security policy on the FortiGate dialup client must refer to the DHCP server address. The DHCP server must be configured to assign a range of IP addresses different from the DHCP server's local network, and also different from the private network addresses behind the FortiGate dialup server. See Routing on page 1.

FortiGate dialup-client infrastructure requirements

The requirements are:

- The FortiGate dialup server must have a static public IP address.
- NAT mode is required if you want to create a route-based VPN.
- The FortiGate dialup server may operate in either NAT mode or transparent mode to support a policy-based VPN.
- Computers on the private network behind the FortiGate dialup client can obtain IP addresses either from a DHCP server behind the FortiGate dialup client, or a DHCP server behind the FortiGate dialup server.
  - If the DHCP server resides on the network behind the dialup client, the DHCP server must be configured to assign IP addresses that do not match the private network behind the FortiGate dialup server.
  - If the DHCP server resides on the network behind the FortiGate dialup server, the DHCP server must be configured to assign IP addresses that do not match the private network behind the FortiGate dialup client.

Configuring the server to accept FortiGate dialup-client connections

The procedures in this section assume that computers on the private network behind the FortiGate dialup client obtain IP addresses from a local DHCP server. The assigned IP addresses do not match the private network behind the FortiGate dialup server.

In situations where IP-address overlap between the local and remote private networks is likely to occur, FortiGate DHCP relay can be configured on the FortiGate dialup client to relay DHCP requests to a DHCP server behind the FortiGate dialup server. For more information, see To configure DHCP relay on a FortiGate interface on page 1.

Configuring dialup client capability for FortiGate dialup clients involves the following general configuration steps:

- Determine which IP addresses to assign to the private network behind the FortiGate dialup client, and add the IP addresses to the DHCP server behind the FortiGate dialup client. Refer to the software supplier's documentation to configure the DHCP server.
- Configure the FortiGate dialup server. See Configuration overview on page 137.
- Configure the FortiGate dialup client. See Configuration overview on page 137.

Before you begin, optionally reserve a unique identifier (peer ID) for the FortiGate dialup client. The dialup client will supply this value to the FortiGate dialup server for authentication purposes during the IPsec Phase 1 exchange. In addition, the value will enable you to distinguish FortiGate dialup-client connections from FortiClient dialup-client connections. The same value must be specified on the dialup server and on the dialup client.

In circumstances where multiple remote dialup VPN tunnels exist, each tunnel must have a peer ID set.

At the FortiGate dialup server, define the Phase 1 parameters needed to authenticate the FortiGate dialup client and establish a secure connection. See Phase 1 parameters on page 46.

1. Go to **VPN > IPsec Tunnels** and create the new custom tunnel or edit an existing tunnel.
2. Edit **Network** (full configuration options are only available once you click the **Convert To Custom Tunnel** button).
3. Enter these settings in particular:

Remote Gateway	Select **Dialup User**.
Interface	Select the interface through which clients connect to the FortiGate unit.

4. Edit **Authentication** and enter the following information:

Mode	If you will be assigning an ID to the FortiGate dialup client, select **Aggressive**.

Peer Options	If you will be assigning an ID to the FortiGate dialup client, set **Accept Types** to **This peer ID** and type the identifier that you reserved for the FortiGate dialup client into the adjacent field.

5. Define the Phase 2 parameters needed to create a VPN tunnel with the FortiGate dialup client. See Phase 2 parameters on page 66. Enter these settings in particular:

Name	Enter a name to identify this Phase 2 configuration.
Phase 1	Select the name of the Phase 1 configuration that you defined.

6. Define names for the addresses or address ranges of the private networks that the VPN links. See Defining policy addresses on page 1. Enter these settings in particular:
   - Define an address name for the server, host, or network behind the FortiGate dialup server.
   - Define an address name for the private network behind the FortiGate dialup client.
4. Define the security policies to permit communications between the private networks through the VPN tunnel. Route-based and policy-based VPNs require different security policies. For detailed information about creating security policies, see Defining VPN security policies on page 1.

Route-based VPN security policy

Define an ACCEPT security policy to permit communications between hosts on the private network behind the FortiGate dialup client and the private network behind this FortiGate dialup server. Because communication cannot be initiated in the opposite direction, there is only one policy.

1. Go to **Policy & Objects > IPv4 Policy** and select **Create New**.
2. Enter these settings in particular:

Name	Enter an appropriate name for the policy.
Incoming Interface	Select the VPN tunnel (IPsec interface) created in Step 1.
Outgoing Interface	Select the interface that connects to the private network behind this FortiGate unit.
Source	Select **all**.
Destination Address	Select **all**.
Action	Select **ACCEPT**.
NAT	Disable **NAT**.

Policy-based VPN security policy

1. Go to **Policy & Objects > IPv4 Policy** and select **Create New**.
2. Enter these settings in particular:

Name	Enter an appropriate name for the policy.

Incoming Interface	Select the interface that connects to the private network behind this FortiGate unit.
Outgoing Interface	Select the FortiGate unit's public interface.
Source	Select the address name that you defined for the private network behind this FortiGate unit.
Destination Address	Select the address name that you defined.
Action	Select **IPsec**. Under **VPN Tunnel**, select the name of the Phase 1 configuration that you created in Step "Configuration overview " on page 137 from the drop-down list. Select **Allow traffic to be initiated from the remote site**.

3. To prevent traffic from the local network from initiating the tunnel after the tunnel has been established, you need to disable the outbound VPN traffic in the CLI

```
config firewall policy
   edit <policy_number>
      set outbound disable
end
```

Place the policy in the policy list above any other policies having similar source and destination addresses.

If configuring a route-based policy, configure a default route for VPN traffic on this interface.

Configuring the FortiGate dialup client

At the FortiGate dialup client, define the Phase 1 parameters needed to authenticate the dialup server and establish a secure connection. See Phase 1 parameters on page 46.

1. Go to **VPN > IPsec Tunnels** and create the new custom tunnel or edit an existing tunnel.
2. Edit **Network** (full configuration options are only available once you click the **Convert To Custom Tunnel** button).
3. Enter these settings in particular:

Remote Gateway	Select **Static IP Address**.
IP Address	Type the IP address of the dialup server's public interface.
Interface	Select the interface that connects to the public network.
Mode	The FortiGate dialup client has a dynamic IP address, select **Aggressive**.
Advanced	Select to view the following options.
Local ID	If you defined a peer ID for the dialup client in the FortiGate dialup server configuration, enter the identifier of the dialup client. The value must be identical to the peer ID that you specified previously in the FortiGate dialup server configuration.

4. Edit **Authentication** and enter the following information:

Mode	The FortiGate dialup client has a dynamic IP address, select **Aggressive**.

5. Edit **Phase 1 Proposal** and enter the following information:

Local ID	If you defined a peer ID for the dialup client in the FortiGate dialup server configuration, enter the identifier of the dialup client. The value must be identical to the peer ID that you specified previously in the FortiGate dialup server configuration.

6. Define the Phase 2 parameters needed to create a VPN tunnel with the dialup server. See Phase 2 parameters on page 66. Enter these settings in particular:

Name	Enter a name to identify this Phase 2 configuration.
Phase 1	Select the name of the Phase 1 configuration that you defined.

7. Define names for the addresses or address ranges of the private networks that the VPN links. See Defining policy addresses on page 1. Enter these settings in particular:
   - Define an address name for the server, host, or network behind the FortiGate dialup server.
   - Define an address name for the private network behind the FortiGate dialup client.
4. Define security policies to permit communication between the private networks through the VPN tunnel. Route-based and policy-based VPNs require different security policies. For detailed information about creating security policies, see Defining VPN security policies on page 1.

Route-based VPN security policy

Define an ACCEPT security policy to permit communications between hosts on the private network behind this FortiGate dialup client and the private network behind the FortiGate dialup server. Because communication cannot be initiated in the opposite direction, there is only one policy.

1. Go to **Policy & Objects > IPv4 Policy** and select **Create New**.
2. Enter these settings in particular:

Name	Enter an appropriate name for the policy.
Incoming Interface	Select the interface that connects to the private network behind this FortiGate unit.
Outgoing Interface	Select the VPN tunnel (IPsec interface) created in Step 1.
Source	Select **all**.
Destination Address	Select **all**.
Action	Select **ACCEPT**.
NAT	Disable **NAT**.

Policy-based VPN security policy

Define an IPsec security policy to permit communications between the source and destination addresses.

1. Go to **Policy & Objects > IPv4 Policy** and select **Create New**.
2. Enter these settings in particular:

Incoming Interface	Select the interface that connects to the private network behind this FortiGate unit.
Outgoing Interface	Select the FortiGate unit's public interface.
Source	Select the address name that you defined for the private network behind this FortiGate unit.
Destination Address	Select the address name that you defined for the private network behind the dialup server.
Action	Select **IPsec**. Under **VPN Tunnel**, select the name of the Phase 1 configuration that you created in Step "Configuration overview " on page 137 from the drop-down list. Clear **Allow traffic to be initiated from the remote site** to prevent traffic from the remote network from initiating the tunnel after the tunnel has been established.

Place the policy in the policy list above any other policies having similar source and destination addresses.

IPsec dial-up interface sharing

It is possible to use a single interface for all instances that spawn via a given phase1. In this case, instead of creating an interface per instance, all traffic will run over the single interface and any routes that need creating will be created on that single interface.

The CLI option "`net-device[enable|disable]`" is available in the phase1-interface command sets. Under the new single-interface scheme, instead of relying on routing to guide traffic to the specific instance, all traffic will flow to the specific device and IPsec will need to take care of locating the correct instance for outbound traffic. For this purpose, the CLI option "tunnel-search" is provided. The option is only available when the above "net-device" option is "disable".

There are two options for "tunnel-search", corresponding to the two ways to select the tunnel for outbound traffic. One is "selectors", meaning selecting a peer using the IPsec selectors (proxy-ids). The other is "nexthop" where all the peers use the same default selectors (0/0) while using some routing protocols such as BGP, OSPF, RIPng, etc to resolve the routing. The default for "tunnel-search" is "selectors".

Syntax

```
config vpn ipsec phase1-interface
   edit xxx
      set net-device [enable|disable]   Enable to create a kernel device for every dialup instance
   next
end
config vpn ipsec phase1-interface
   edit xxx
      set net-device disable
      set tunnel-search [selectors|nexthop]   Search for tunnel in selectors or using nexthops
   next
end
```

Supporting IKE Mode Config clients

IKE Mode Config is an alternative to DHCP over IPsec. A FortiGate unit can be configured as either an IKE Mode Config server or client. This chapter contains the following sections:

IKE Mode Config overview

Dialup VPN clients connect to a FortiGate unit that acts as a VPN server, providing the client the necessary configuration information to establish a VPN tunnel. The configuration information typically includes a virtual IP address, netmask, and DNS server address.

IKE Mode Config is available only for VPNs that are route-based, also known as interface-based. A FortiGate unit can function as either an IKE Configuration Method server or client. IKE Mode Config is configurable only in the CLI.

Automatic configuration overview

VPN configuration for remote clients is simpler if it is automated. Several protocols support automatic configuration:

- The Fortinet FortiClient Endpoint Security application can completely configure a VPN connection with a suitably configured FortiGate unit given only the FortiGate unit's address. This protocol is exclusive to Fortinet. For more information, see FortiClient dialup-client configurations on page 1.
- DHCP over IPsec can assign an IP address, Domain, DNS and WINS addresses. The user must first configure IPsec parameters such as gateway address, encryption and authentication algorithms.
- IKE Mode Config can configure host IP address, Domain, DNS and WINS addresses. The user must first configure IPsec parameters such as gateway address, encryption and authentication algorithms. Several network equipment vendors support IKE Mode Config, which is described in the ISAKMP Configuration Method document draft-dukes-ike-mode-cfg-02.txt.

This chapter describes how to configure a FortiGate unit as either an IKE Mode Config server or client.

IKE Mode Config method

IKE Mode Config is configured with the CLI command `config vpn ipsec phase1-interface`. The `mode-cfg` variable enables IKE Mode Config. The `type` field determines whether you are creating an IKE Mode Config server or a client. Setting `type` to `dynamic` creates a server configuration, otherwise the configuration is a client.

Creating an IKE Mode Config client

If the FortiGate unit will connect as a dialup client to a remote gateway that supports IKE Mode Config, the relevant `vpn ipsec phase1-interface` variables are as follows:

Variable	Description
`ike-version 1`	IKE v1 is the default for FortiGate IPsec VPNs. IKE Mode Config is also compatible with IKE v2 (RFC 4306). Use syntax `ike-version 2`.
`mode-cfg enable`	Enable IKE Mode Config.
`type {ddns | static}`	If you set `type` to `dynamic`, an IKE Mode Config server is created.
`assign-ip {enable | disable}`	Enable to request an IP address from the server.
`interface <interface_name>`	This is a regular IPsec VPN field. Specify the physical, aggregate, or VLAN interface to which the IPsec tunnel will be bound.
`proposal <encryption_combination>`	This is a regular IPsec VPN field that determines the encryption and authentication settings that the client will accept. For more information, see Phase 1 parameters on page 46.
`ip-version <4 | 6>`	This is a regular IPsec VPN field. By default, IPsec VPNs use IPv4 addressing. You can set `ip-version` to `6` to create a VPN with IPv6 addressing.
`ipv4-split-exclude` `ipv6-split-exclude`	This command allows the administrator to specify that default traffic flows over the IPsec tunnel except for specified subnets. This is the opposite of the supported `split-include` feature which allows the administrator to specify that default traffic should not flow over the IPsec tunnel except for specified subnets.

For a complete list of available variables, see the CLI Reference.

IKE Mode Config client example - CLI

In this example, the FortiGate unit connects to a VPN gateway with a static IP address that can be reached through Port 1. Only the port, gateway and proposal information needs to be configured. All other configuration information will come from the IKE Mode Config server.

```
config vpn ipsec phase1-interface
   edit vpn1
      set ip-version 4
      set type static
      set remote-gw <gw_address>
      set interface port 1
      set proposal 3des-sha1 aes128-sha1
      set mode-cfg enable
```

```
    set assign-ip enable
end
```

Creating an IKE Mode Config server

If the FortiGate unit will accept connection requests from dialup clients that support IKE Mode Config, the following `vpn ipsec phase1-interface` settings are required before any other configuration is attempted:

Variable	Description
`ike-version 1`	IKE v1 is the default for FortiGate IPsec VPNs. IKE Mode Config is also compatible with IKE v2 (RFC 4306). Use syntax `ike-version 2`.
`mode-cfg enable`	Enable IKE Mode Config.
`type dynamic`	Any other setting creates an IKE Mode Config client.
`interface <interface_name>`	This is a regular IPsec VPN field. Specify the physical, aggregate, or VLAN interface to which the IPsec tunnel will be bound.
`proposal <encryption_combination>`	This is a regular IPsec VPN field that determines the encryption and authentication settings that the server will accept. For more information, see Phase 1 parameters on page 46.
`ip-version <4 | 6>`	This is a regular IPsec VPN field. By default, IPsec VPNs use IPv4 addressing. You can set `ip-version` to `6` to create a VPN with IPv6 addressing.

IKE Mode Config server example - CLI

In this example, the FortiGate unit assigns IKE Mode Config clients addresses in the range of 10.11.101.160 through 10.11.101.180. DNS and WINS server addresses are also provided. The public interface of the FortiGate unit is Port 1.

When IKE Mode-Configuration is enabled, multiple server IPs can be defined in IPsec Phase 1.

The `ipv4-split-include` variable specifies a firewall address that represents the networks to which the clients will have access. This destination IP address information is sent to the clients.

Only the CLI fields required for IKE Mode Config are shown here. For detailed information about these variables, see the FortiGate CLI Reference.

```
config vpn ipsec phase1-interface
edit "vpn-p1"
   set type dynamic
   set interface "wan1"
   set xauthtype auto
   set mode aggressive
   set mode-cfg enable
   set proposal 3des-sha1 aes128-sha1
   set dpd disable
   set dhgrp 2
```

```
    set xauthexpire on-rekey
    set authusrgrp "FG-Group1"
    set ipv4-start-ip 10.10.10.10
    set ipv4-end-ip 10.10.10.20
    set ipv4-dns-server1 1.1.1.1
    set ipv4-dns-server2 2.2.2.2
    set ipv4-dns-server3 3.3.3.3
    set ipv4-wins-server1 4.4.4.4
    set ipv4-wins-server2 5.5.5.5
    set domain "fgt1c-domain"
    set banner "fgt111C-banner"
    set backup-gateway "100.100.100.1" "host1.com" "host2"
    set ipv4-split-include OfficeLAN
end
```

IP address assignment

After you have enabled the basic configuration, you can configure IP address assignment for clients, as well as DNS and WINS server assignment. Usually you will want to assign IP addresses to clients.

The simplest method to assign IP addresses to clients is to assign addresses from a specific range, similar to a DHCP server.

If your clients are authenticated by a RADIUS server, you can obtain the user's IP address assignment from the Framed-IP-Address attribute. The user must be authenticated using XAuth.

IKE Mode Config can also use a remote DHCP server to assign the client IP addresses. Up to eight addresses can be selected for either IPv4 or IPv6. After the DHCP proxy has been configured, the `assign-ip-from` command is used to assign IP addresses via DHCP.

Assigning IP addresses from an address range - CLI

If your VPN uses IPv4 addresses,

```
config vpn ipsec phase1-interface
   edit vpn1
      set mode-cfg-ipversion 4
      set assign-ip enable
      set assign-ip-type ip
      set assign-ip-from range
      set ipv4-start-ip <range_start>
      set ipv4-end-ip <range_end>
      set ipv4-netmask <netmask>
   end
```

If your VPN uses IPv6 addresses,

```
config vpn ipsec phase1-interface
   edit vpn1
      set mode-cfg-ipversion 6
      set assign-ip enable
      set assign-ip-type ip
      set assign-ip-from range
      set ipv6-start-ip <range_start>
      set ipv6-end-ip <range_end>
   end
```

Assigning IP addresses from a RADIUS server - CLI

The users must be authenticated by a RADIUS server and assigned to the FortiGate user group `<grpname>`. Since the IP address will not be static, `type` is set to dynamic, and `mode-cfg` is enabled. This is IKE Configuration Method so that compatible clients can configure themselves with settings that the FortiGate unit provides.

```
config vpn ipsec phase1-interface
   edit vpn1
      set type dynamic
         set mode-cfg enable
         set assign-ip enable
         set assign-ip-from usrgrp
         set xauthtype auto
         set authusrgrp <grpname>
   end
```

Assigning IP address from DHCP - CLI

The DHCP proxy must first be enabled for IKE Mode Config to use DHCP to assign the VPN client IP address(es).

```
config system settings
   set dhcp-proxy enable
   set dhcp-server-ip [ipv4 address]
   set dhcp6-server-ip [ipv6-address]
```

(Up to eight server addresses can be configured)

```
end

config vpn ipsec phase1-interface
   edit vpn1
      set mode-cfg enable
      set assign-ip-from dhcp
   next
end
```

Assigning IP address from a named firewall address/group - CLI

```
config vpn ipsec phase1-interface
   edit <name>vpn1
      set type dynamic
      set assign-ip-from name
      set ipv4-name <name>
      set ipv6-name <name>
   next
end
```

Certificate groups

IKE certificate groups consisting of up to four RSA certificates can be used in IKE Phase 1. Since CA and local certificates are global, the IKE daemon loads them once for all VDOMs and indexes them into trees based on subject and public key hash (for CA certificates), or certificate name (for local certicates). Certifcates are linked together based on the issuer, and certificate chains are built by traversing these links. This reduces the need to keep multiple copies of certificates that could exist in multiple chains.

IKE certificate groups can be configured through the CLI.

Configuring the IKE local ID - CLI

```
config vpn certificate local
edit <name>
   set ike-localid <string>
   set ike-localid-type {asnldn | fqdn}
end
```

Split-exclude in IKEv1 mode-cfg

This feature allows the administrator to specify when using IKEv1 Configuration Method that default traffic flows over the IPsec tunnel except for specified subnets. This is the opposite of the supported `split-include` feature which allows the administrator to specify that default traffic should not flow over the IPsec tunnel except for specified subnets.

The `split-include` and `split-exclude` options can both be specified at the same time. Whether a client does the right thing when both are specified depends on the client.

Syntax

```
config vpn ipsec {phase1 | phase1-interface}
   edit <name>
      set ike-version 1
      set type dynamic
      set mode-cfg enable
      set ipv4-split-exclude {all | none | address}
      set ipv6-split-exclude {all | none | address}
   next
end
```

Internet-browsing configuration

This section explains how to support secure web browsing performed by dialup VPN clients, and/or hosts behind a remote VPN peer. Remote users can access the private network behind the local FortiGate unit and browse the Internet securely. All traffic generated remotely is subject to the security policy that controls traffic on the private network behind the local FortiGate unit.

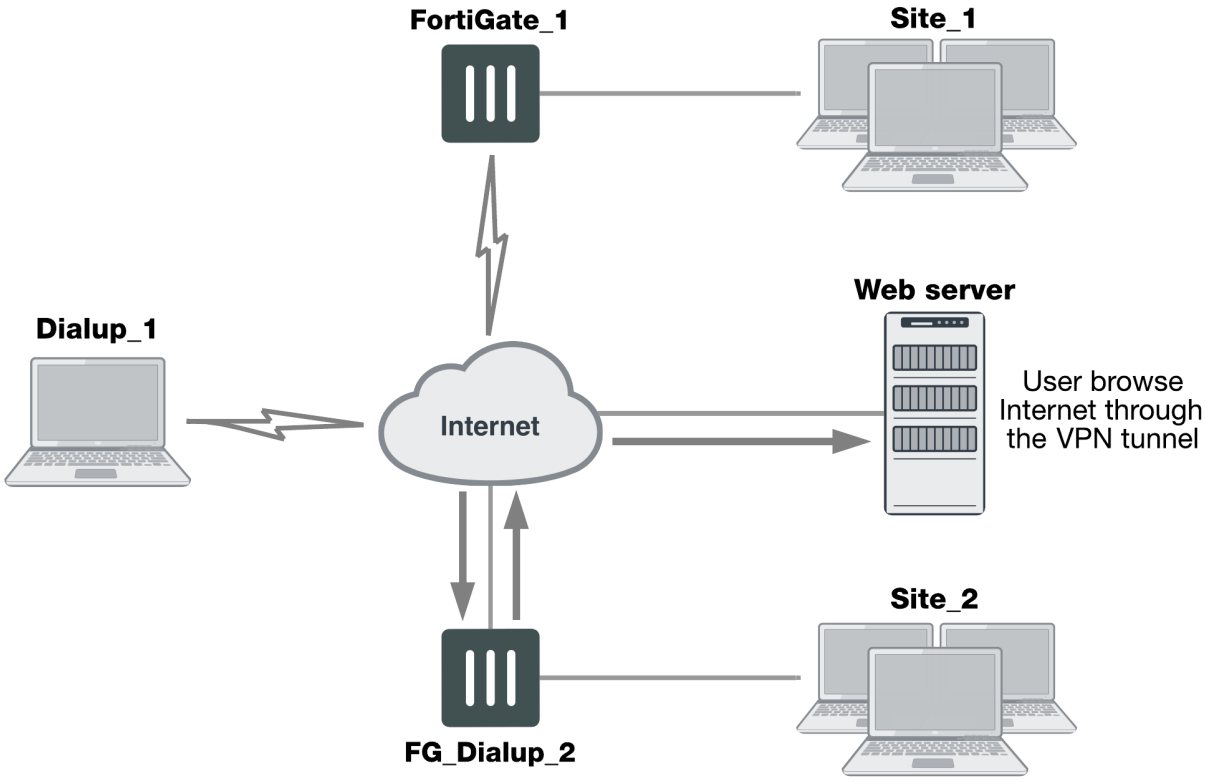
The following topics are included in this section:

Configuration overview

A VPN provides secure access to a private network behind the FortiGate unit. You can also enable VPN clients to access the Internet securely. The FortiGate unit inspects and processes all traffic between the VPN clients and hosts on the Internet according to the Internet browsing policy. This is accomplished even though the same FortiGate interface is used for both encrypted VPN client traffic and unencrypted Internet traffic.

In the figure below, FortiGate_1 enables secure Internet browsing for FortiClient Endpoint Security users such as Dialup_1 and users on the Site_2 network behind FortiGate_2, which could be a VPN peer or a dialup client.

Example Internet-browsing configuration



You can adapt any of the following configurations to provide secure Internet browsing:

- A gateway-to-gateway configuration (see Gateway-to-gateway configurations on page 1)
- A FortiClient dialup-client configuration (see FortiClient dialup-client configurations on page 1)
- A FortiGate dialup-client configuration (see FortiGate dialup-client configurations on page 1)

The procedures in this section assume that one of these configurations is in place, and that it is operating properly.

To create an internet-browsing configuration based on an existing gateway-to-gateway configuration, you must edit the gateway-to-gateway configuration as follows:

- On the FortiGate unit that will provide Internet access, create an Internet browsing security policy. See Configuration overview on page 151, below.
- Configure the remote peer or client to route all traffic through the VPN tunnel. You can do this on a FortiGate unit or on a FortiClient Endpoint Security application. See Configuration overview on page 151.

Creating an Internet browsing security policy

On the FortiGate unit that acts as a VPN server and will provide secure access to the Internet, you must create an Internet browsing security policy. This policy differs depending on whether your gateway-to-gateway configuration is policy-based or route-based.

Creating an Internet browsing policy - policy-based VPN

1. Go to **Policy & Objects > IPv4 Policy** and select **Create New**.
2. Enter the following information and then select **OK**:

Name	Enter an appropriate name for the policy.
Incoming Interface	The interface to which the VPN tunnel is bound.
Outgoing Interface	The interface to which the VPN tunnel is bound.
Source	The internal range address of the remote spoke site.
Destination Address	**all**
Action	Select **IPsec**. Under **VPN Tunnel**, select the tunnel that provides access to the private network behind the FortiGate unit. Select **Allow traffic to be initiated from the remote site**.
NAT	Enable **NAT**.

Creating an Internet browsing policy - route-based VPN

1. Go to **Policy & Objects > IPv4 Policy** and select **Create New**.
2. Enter the following information and then select **OK**:

Name	Enter an appropriate name for the policy.

Incoming Interface	The IPsec VPN interface.
Outgoing Interface	The interface that connects to the Internet. The virtual IPsec interface is configured on this physical interface.
Source	The internal range address of the remote spoke site.
Destination Address	**all**
Action	ACCEPT
NAT	Enable **NAT**.

The VPN clients must be configured to route all Internet traffic through the VPN tunnel.

Routing all remote traffic through the VPN tunnel

To make use of the Internet browsing configuration on the VPN server, the VPN peer or client must route all traffic through the VPN tunnel. Usually, only the traffic destined for the private network behind the FortiGate VPN server is sent through the tunnel.

The remote end of the VPN can be a FortiGate unit that acts as a peer in a gateway-to-gateway configuration, or a FortiClient application that protects an individual client PC.

- To configure a remote peer FortiGate unit for Internet browsing via VPN, see Configuring a FortiGate remote peer to support Internet browsing on page 153.
- To configure a FortiClient Endpoint Security application for Internet browsing via VPN, see Configuring a FortiClient application to support Internet browsing on page 154.

These procedures assume that your VPN connection to the protected private network is working and that you have configured the FortiGate VPN server for Internet browsing as described in Configuration overview on page 151.

Configuring a FortiGate remote peer to support Internet browsing

The configuration changes to send all traffic through the VPN differ for policy-based and route-based VPNs.

Routing all traffic through a policy-based VPN

1. At the FortiGate dialup client, go to **Policy & Objects > IPv4 Policy**.
2. Select the IPsec security policy and then select **Edit**.
3. From the **Destination Address** list, select **all**.
4. Select **OK**.

Packets are routed through the VPN tunnel, not just those destined for the protected private network.

Routing all traffic through a route-based VPN

1. At the FortiGate dialup client, go to **Network > Static Routes**.
2. Select the default route (destination IP 0.0.0.0) and then select **Edit**. If there is no default route, select **Create New**. Enter the following information and select **OK**:

Destination IP/Mask	Set to **Subnet** and enter **0.0.0.0/0.0.0.0** in the field provided.
Device	Select the IPsec virtual interface.
Administrative Distance	Leave at default.

All packets are routed through the VPN tunnel, not just packets destined for the protected private network.

Configuring a FortiClient application to support Internet browsing

By default, the FortiClient application configures the PC so that traffic destined for the remote protected network passes through the VPN tunnel but all other traffic is sent to the default gateway. You need to modify the FortiClient settings so that it configures the PC to route all outbound traffic through the VPN.

Routing all traffic through VPN - FortiClient application

1. At the remote host, start FortiClient.
2. Go to **Remote Access**.
3. Select the definition that connects FortiClient to the FortiGate dialup server, select the **Settings** icon, and select **Edit the selected connection**.
4. In the **Edit VPN Connection** dialog box, select **Advanced Settings**.
5. In the **Remote Network** group, select **Add**.
6. In the **IP** and **Subnet Mask** fields, type `0.0.0.0/0.0.0.0` and select **OK**.

   The address is added to the **Remote Network** list. The first destination IP address in the list establishes a VPN tunnel. The second destination address (`0.0.0.0/0.0.0.0` in this case) forces all other traffic through the VPN tunnel.
7. Select **OK**.

Redundant VPN configurations

This section discusses the options for supporting redundant and partially redundant IPsec VPNs, using route-based approaches.

The following topics are included in this section:

Configuration overview

A FortiGate unit with two interfaces connected to the Internet can be configured to support redundant VPNs to the same remote peer. If the primary connection fails, the FortiGate unit can establish a VPN using the other connection.

Redundant tunnels do not support Tunnel Mode or manual keys. You must use Interface Mode.

A fully-redundant configuration requires redundant connections to the Internet on both peers. The figure below shows an example of this. This is useful to create a reliable connection between two FortiGate units with static IP addresses.

When only one peer has redundant connections, the configuration is partially-redundant. For an example of this, see Configuration overview on page 155. This is useful to provide reliable service from a FortiGate unit with static IP addresses that accepts connections from dialup IPsec VPN clients.

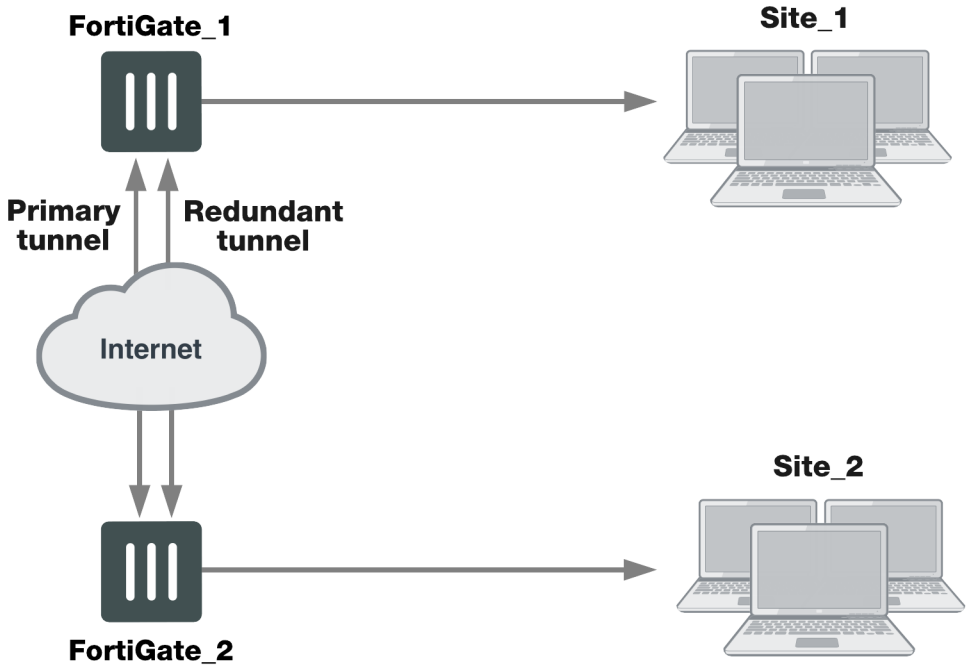
In a fully-redundant VPN configuration with two interfaces on each peer, four distinct paths are possible for VPN traffic from end to end. Each interface on a peer can communicate with both interfaces on the other peer. This ensures that a VPN will be available as long as each peer has one working connection to the Internet.

You configure a VPN and an entry in the routing table for each of the four paths. All of these VPNs are ready to carry data. You set different routing distances for each route and only the shortest distance route is used. If this route fails, the route with the next shortest distance is used.

The redundant configurations described in this chapter use route-based VPNs, otherwise known as virtual IPsec interfaces. This means that the FortiGate unit must operate in NAT mode. You must use auto-keying. A VPN that is created using manual keys cannot be included in a redundant-tunnel configuration.

The configuration described here assumes that your redundant VPNs are essentially equal in cost and capability. When the original VPN returns to service, traffic continues to use the replacement VPN until the replacement VPN fails. If your redundant VPN uses more expensive facilities, you want to use it only as a backup while the main VPN is down. For information on how to do this, see Configuration overview on page 155.

Example redundant-tunnel configuration



A VPN that is created using manual keys cannot be included in a redundant-tunnel configuration.

General configuration steps

A redundant configuration at each VPN peer includes:

- One Phase 1 configuration (virtual IPsec interface) for each path between the two peers. In a fully-meshed redundant configuration, each network interface on one peer can communicate with each network interface on the remote peer. If both peers have two public interfaces, this means that each peer has four paths, for example.
- One Phase 2 definition for each Phase 1 configuration.
- One static route for each IPsec interface, with different distance values to prioritize the routes.
- Two Accept security policies per IPsec interface, one for each direction of traffic.
- Dead peer detection enabled in each Phase 1 definition.

The procedures in this section assume that two separate interfaces to the Internet are available on each VPN peer.

Configuring the VPN peers - route-based VPN

VPN peers are configured using Interface Mode for redundant tunnels.

Configure each VPN peer as follows:

1. Ensure that the interfaces used in the VPN have static IP addresses.
2. Create a Phase 1 configuration for each of the paths between the peers.
3. Enable dead peer detection so that one of the other paths is activated if this path fails.
4. Enter these settings in particular, and any other VPN settings as required:

Path 1

Remote Gateway	Select **Static IP Address**.
IP Address	Type the IP address of the primary interface of the remote peer.
Local Interface	Select the primary public interface of this peer.
Dead Peer Detection	Enable

Path 2

Remote Gateway	Select **Static IP Address**.
IP Address	Type the IP address of the secondary interface of the remote peer.
Local Interface	Select the primary public interface of this peer.
Dead Peer Detection	Enable

Path 3

Remote Gateway	Select **Static IP Address**.
IP Address	Type the IP address of the primary interface of the remote peer.
Local Interface	Select the secondary public interface of this peer.
Dead Peer Detection	Enable

Path 4

Remote Gateway	Select **Static IP Address**.
IP Address	Type the IP address of the secondary interface of the remote peer.
Local Interface	Select the secondary public interface of this peer.
Dead Peer Detection	Enable

For more information, see Phase 1 parameters on page 46.

5. Create a Phase 2 definition for each path. See Phase 2 parameters on page 66. Select the Phase 1 configuration (virtual IPsec interface) that you defined for this path. You can select the name from the Static IP Address part of the list.
6. Create a route for each path to the other peer. If there are two ports on each peer, there are four possible paths between the peer devices.

Destination IP/Mask	The IP address and netmask of the private network behind the remote peer.
Device	One of the virtual IPsec interfaces on the local peer.
Distance	For each path, enter a different value to prioritize the paths.

7. Define the security policy for the local primary interface. See Defining VPN security policies on page 1. You need to create two policies for each path to enable communication in both directions. Enter these settings in particular:

Incoming Interface	Select the local interface to the internal (private) network.
Source Address	All
Outgoing Interface	Select one of the virtual IPsec interfaces you created in Step 2.
Destination Address	All
Schedule	Always
Service	Any
Action	ACCEPT

8. Select **Create New**, leave the **Policy Type** as **Firewall** and leave the **Policy Subtype** as **Address**, and enter these settings:

Incoming Interface	Select one of the virtual IPsec interfaces you created in Step 2.
Source Address	All
Outgoing Interface	Select the local interface to the internal (private) network.
Destination Address	All
Schedule	Always
Service	Any
Action	ACCEPT

9. Place the policy in the policy list above any other policies having similar source and destination addresses.
10. Repeat this procedure at the remote FortiGate unit.

Creating a backup IPsec interface

You can configure a route-based VPN that acts as a backup facility to another VPN. It is used only while your main VPN is out of service. This is desirable when the redundant VPN uses a more expensive facility.

You can configure a backup IPsec interface only in the CLI. The backup feature works only on interfaces with static addresses that have dead peer detection enabled. The `monitor` option creates a backup VPN for the specified Phase 1 configuration.

In the following example, backup_vpn is a backup for main_vpn.

```
config vpn ipsec phase1-interface
   edit main_vpn
      set dpd on
      set interface port1
      set nattraversal enable
      set psksecret "hard-to-guess"
      set remote-gw 192.168.10.8
      set type static
   end
   edit backup_vpn
      set dpd on
      set interface port2
      set monitor main_vpn
      set nattraversal enable
      set psksecret "hard-to-guess"
      set remote-gw 192.168.10.8
      set type static
   end
```

IPsec VPN tunnel aggregate interfaces

This feature allows per-packet routing decisions to be made over two or more IPsec tunnel interfaces, which is usually configured to allow WAN connections to terminate at a data center so that redundancy and load-sharing can be built into this new interface.

The new virtual interface can bond/aggregate IPsec devices and have the new device do round-robin distribution, among other algorithms.

Syntax

```
config vpn ipsec phase1-interface
   edit <name>
      set interface wan1
      set gateway ...
      ...
   next
   edit <name>
      set interface wan2
      set gateway ...
      ...
   next
end
config vpn ipsec phase2-interface
```

```
   ...
end
config system interface
   edit ipsec-bond
      set type tun-agg
      set member isp1 isp2
   next
end
config router static
   edit <value>
      set dst <address>
      set device ipsec-bond
   next
end
```

Transparent mode VPNs

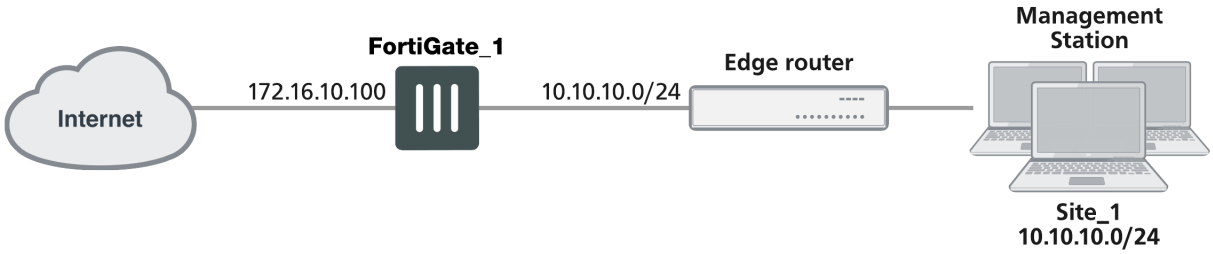
This section describes transparent VPN configurations, in which two FortiGate units create a VPN tunnel between two separate private networks transparently.

The following topics are included in this section:

Configuration overview

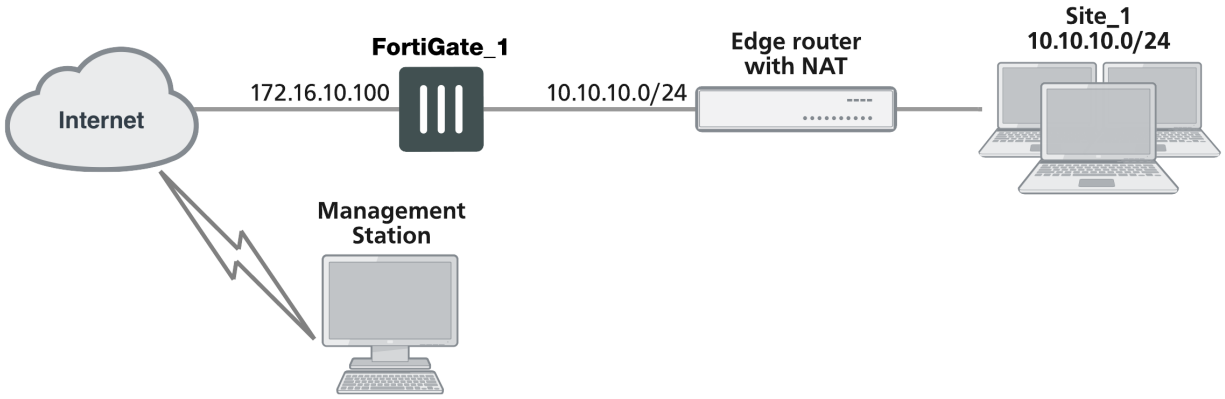
In transparent mode, all interfaces of the FortiGate unit except the management interface (which by default is assigned IP address 10.10.10.1/255.255.255.0) are invisible at the network layer. Typically, when a FortiGate unit runs in transparent mode, different network segments are connected to the FortiGate interfaces. The figure below shows the management station on the same subnet. The management station can connect to the FortiGate unit directly through the web-based manager.

Management station on internal network



An edge router typically provides a public connection to the Internet and one interface of the FortiGate unit is connected to the router. If the FortiGate unit is managed from an external address (see the figure below), the router must translate (NAT) a routable address to direct management traffic to the FortiGate management interface.

Management station on external network



In a transparent VPN configuration, two FortiGate units create a VPN tunnel between two separate private networks transparently. All traffic between the two networks is encrypted and protected by FortiGate security policies.

Both FortiGate units may be running in transparent mode, or one could be running in transparent mode and the other running in NAT mode. If the remote peer is running in NAT mode, it must have a static public IP address.

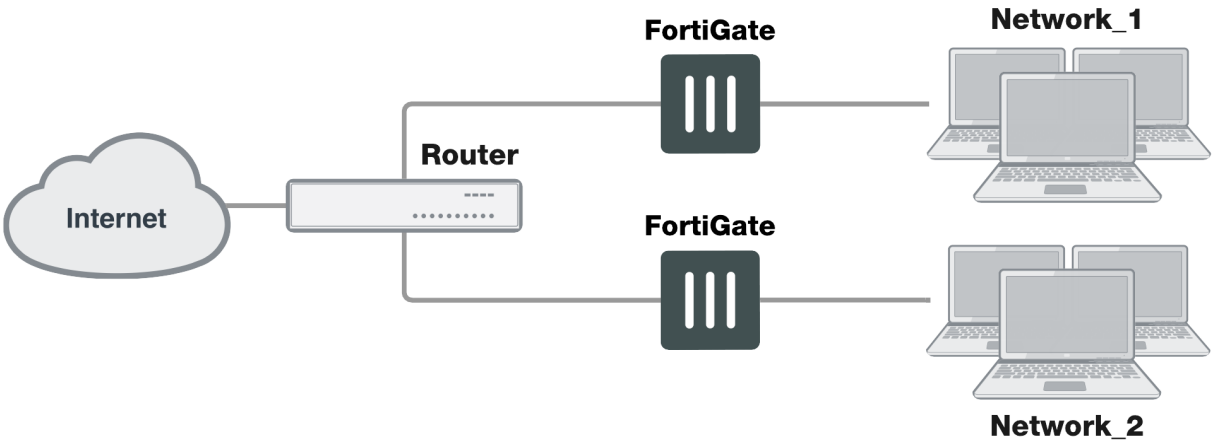
VPNs between two FortiGate units running in transparent mode do not support inbound/outbound NAT (supported through CLI commands) within the tunnel. In addition, a FortiGate unit running in transparent mode cannot be used in a hub-and-spoke configuration.

Encrypted packets from the remote VPN peer are addressed to the management interface of the local FortiGate unit. If the local FortiGate unit can reach the VPN peer locally, a static route to the VPN peer must be added to the routing table on the local FortiGate unit. If the VPN peer connects through the Internet, encrypted packets from the local FortiGate unit must be routed to the edge router instead. For information about how to add a static route to the FortiGate routing table, see the Advanced Routing Guide.

In the example configuration shown above, Network Address Translation (NAT) is enabled on the router. When an encrypted packet from the remote VPN peer arrives at the router through the Internet, the router performs inbound NAT and forwards the packet to the FortiGate unit. Refer to the software supplier's documentation to configure the router.

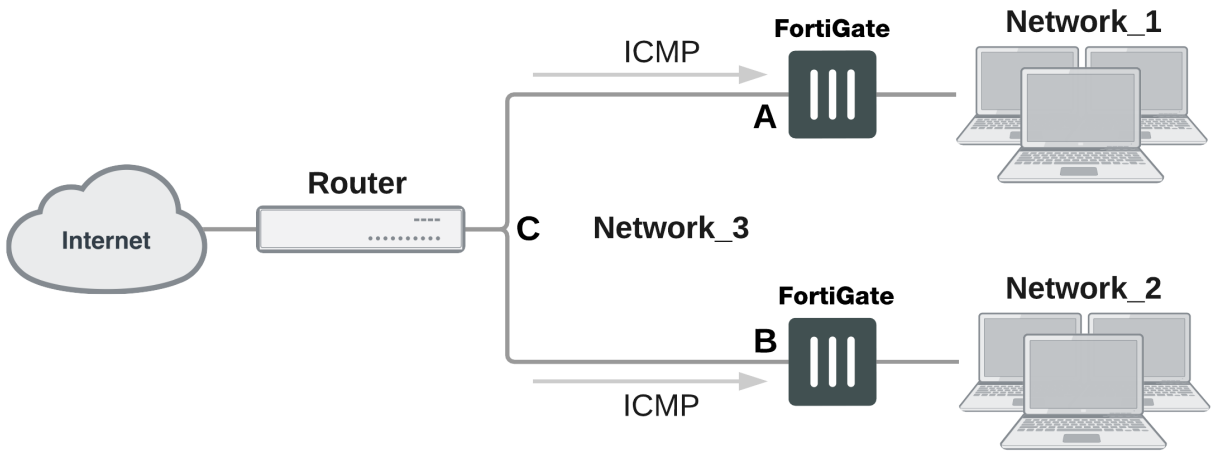
If you want to configure a VPN between two FortiGate units running in transparent mode, each unit must have an independent connection to a router that acts as a gateway to the Internet, and both units must be on separate networks that have a different address space. When the two networks linked by the VPN tunnel have different address spaces (see the figure below), at least one router must separate the two FortiGate units, unless the packets can be redirected using ICMP (as shown in the following figure).

Link between two FortiGate units in transparent mode



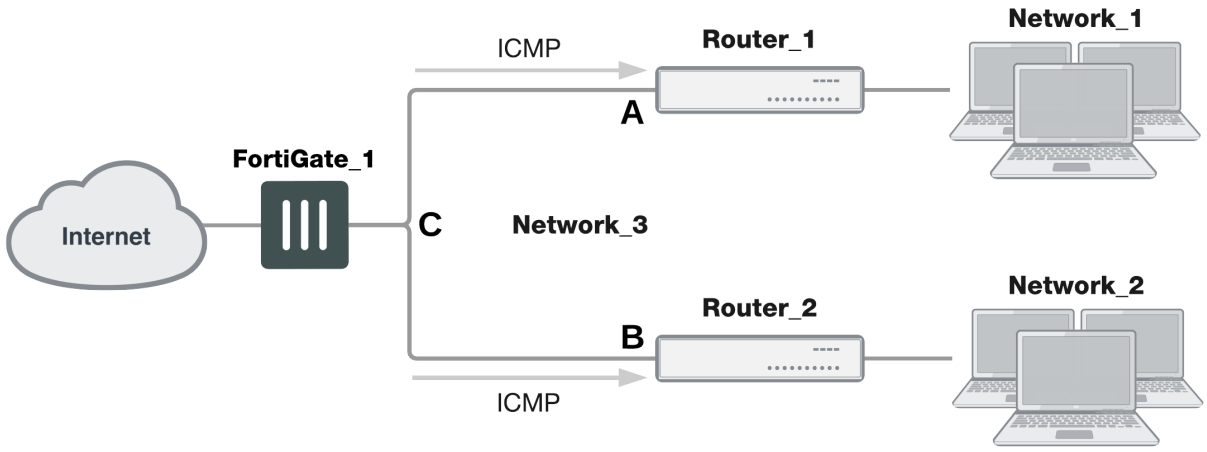
In the figure below, interface C behind the router is the default gateway for both FortiGate units. Packets that cannot be delivered on Network_1 are routed to interface C by default. Similarly, packets that cannot be delivered on Network_2 are routed to interface C. In this case, the router must be configured to redirect packets destined for Network_1 to interface A and redirect packets destined for Network_2 to interface B.

ICMP redirecting packets to two FortiGate units in transparent mode



If there are additional routers behind the FortiGate unit (see the figure below) and the destination IP address of an inbound packet is on a network behind one of those routers, the FortiGate routing table must include routes to those networks. For example, in the following figure, the FortiGate unit must be configured with static routes to interfaces A and B in order to forward packets to Network_1 and Network_2 respectively.

Destinations on remote networks behind internal routers



Transparent VPN infrastructure requirements

- The local FortiGate unit must be operating in transparent mode.
- The management IP address of the local FortiGate unit specifies the local VPN gateway. The management IP address is considered a static IP address for the local VPN peer.
- If the local FortiGate unit is managed through the Internet, or if the VPN peer connects through the Internet, the edge router must be configured to perform inbound NAT and forward management traffic and/or encrypted packets to the FortiGate unit.
- If the remote peer is operating in NAT mode, it must have a static public IP address.

A FortiGate unit operating in transparent mode requires the following basic configuration to operate as a node on the IP network:

- The unit must have sufficient routing information to reach the management station.
- For any traffic to reach external destinations, a default static route to an edge router that forwards packets to the Internet must be present in the FortiGate routing table.
- When all of the destinations are located on the external network, the FortiGate unit may route packets using a single default static route. If the network topology is more complex, one or more static routes in addition to the default static route may be required in the FortiGate routing table.

Only policy-based VPN configurations are possible in transparent mode.

Before you begin

An IPsec VPN definition links a gateway with a tunnel and an IPsec policy. If your network topology includes more than one virtual domain, you must choose components that were created in the same virtual domain. Therefore, before you define a transparent VPN configuration, choose an appropriate virtual domain in which to create the required interfaces, security policies, and VPN components. For more information, see the Virtual Domains guide.

Configuring the VPN peers

1. The local VPN peer need to operate in transparent mode.
   To determine if your FortiGate unit is in transparent mode, go to the **Dashboard > System Information** widget. Select **[change]**. Select transparent for the **Operation Mode**. Two new fields will appear to enter the **Management IP/Netmask**, and the **Default Gateway**.
   In transparent mode, the FortiGate unit is invisible to the network. All of its interfaces are on the same subnet and share the same IP address. You only have to configure a management IP address so that you can make configuration changes.

   The remote VPN peer may operate in NAT mode or transparent mode.
2. At the local FortiGate unit, define the Phase 1 parameters needed to establish a secure connection with the remote peer. See Phase 1 parameters on page 46. Select **Advanced** and enter these settings in particular:

Remote Gateway	Select **Static IP Address**.
IP Address	Type the IP address of the public interface to the remote peer. If the remote peer is a FortiGate unit running in transparent mode, type the IP address of the remote management interface.
Advanced	Select **Nat-traversal**, and type a value into the **Keepalive Frequency** field. These settings protect the headers of encrypted packets from being altered by external NAT devices and ensure that NAT address mappings do not change while the VPN tunnel is open. For more information, see Phase 1 parameters on page 46 and Phase 1 parameters on page 46.

3. Define the Phase 2 parameters needed to create a VPN tunnel with the remote peer. See Phase 2 parameters on page 66. Select the set of Phase 1 parameters that you defined for the remote peer. The name of the remote peer can be selected from the **Static IP Address** list.
4. Define the source and destination addresses of the IP packets that are to be transported through the VPN tunnel. See Defining VPN security policies on page 1. Enter these settings in particular:

- For the originating address (source address), enter the IP address and netmask of the private network behind the local peer network. for the management interface, for example, `10.10.10.0/24`. This address needs to be a range to allow traffic from your network through the tunnel. Optionally select `any` for this address.
- For the remote address (destination address), enter the IP address and netmask of the private network behind the remote peer (for example, `192.168.10.0/24`). If the remote peer is a FortiGate unit running in transparent mode, enter the IP address of the remote management interface instead.

5. Define an IPsec security policy to permit communications between the source and destination addresses. See Defining VPN security policies on page 1. Enter these settings in particular:

Incoming Interface	Select the local interface to the internal (private) network.
Source Address	Select the source address that you defined in Step 4.
Outgoing Interface	Select the interface to the edge router. When you configure the IPsec security policy on a remote peer that operates in NAT mode, you select the public interface to the external (public) network instead.
Destination Address	Select the destination address that you defined in Step 4.
VPN Tunnel	Select **Use Existing** and select the name of the Phase 2 tunnel configuration that you created in Step 3 from the drop-down list. Select **Allow traffic to be initiated from the remote site** to enable traffic from the remote network to initiate the tunnel.

6. Place the policy in the policy list above any other policies having similar source and destination addresses.
7. Define another IPsec security policy to permit communications between the source and destination addresses in the opposite direction. This security policy and the previous one form a bi-directional policy pair. See Defining VPN security policies on page 1. Enter these settings in particular:

Incoming Interface	Select the interface to the edge router. When you configure the IPsec security policy on a remote peer that operates in NAT mode, you select the public interface to the external (public) network instead.
Source Address	Select the destination address that you defined in Step 4..
Outgoing Interface	Select the local interface to the internal (private) network.
Destination Address	Select the source address that you defined in Step 4.
VPN Tunnel	Select **Use Existing** and select the name of the Phase 2 tunnel configuration that you created in Step 3 from the drop-down list. Select **Allow traffic to be initiated from the remote site** to enable traffic from the remote network to initiate the tunnel.

8. Repeat this procedure at the remote FortiGate unit to create bidirectional security policies. Use the local interface and address information local to the remote FortiGate unit.

For more information on transparent mode, see the System Administration Guide.

IPv6 IPsec VPNs

This chapter describes how to configure your FortiGate unit's IPv6 IPsec VPN functionality.

By default IPv6 configurations to not appear on the Web-based Manager. You need to enable the feature first.

To enable IPv6

1. Go to **System > Feature Visibility**.
2. Enable **IPv6**.
3. Select **Apply**.

The following topics are included in this section:

IPv6 IPsec support

FortiOS supports route-based IPv6 IPsec, but not policy-based. This section describes how IPv6 IPsec support differs from IPv4 IPsec support. FortiOS 4.0 MR3 is IPv6 Ready Logo Program Phase 2 certified.

Where both the gateways and the protected networks use IPv6 addresses, sometimes called IPv6 over IPv6, you can create either an auto-keyed or manually-keyed VPN. You can combine IPv6 and IPv4 addressing in an auto-keyed VPN in the following ways:

IPv4 over IPv6	The VPN gateways have IPv6 addresses. The protected networks have IPv4 addresses. The Phase 2 configurations at either end use IPv4 selectors.
IPv6 over IPv4	The VPN gateways have IPv4 addresses. The protected networks use IPv6 addresses. The Phase 2 configurations at either end use IPv6 selectors.

Compared with IPv4 IPsec VPN functionality, there are some limitations:

- Except for IPv6 over IPv4, remote gateways with Dynamic DNS are not supported.
- Selectors cannot be firewall address names. Only IP address, address range and subnet are supported.
- Redundant IPv6 tunnels are not supported.

Certificates

On a VPN with IPv6 Phase 1 configuration, you can authenticate using VPN certificates in which the common name (cn) is an IPv6 address. The `cn-type` keyword of the `user peer` command has an option, `ipv6`, to support this.

Configuration examples

This section consists of the following configuration examples:

- Site-to-site IPv6 over IPv6 VPN example
- Site-to-site IPv6 over IPv4 VPN example
- Site-to-site IPv4 over IPv6 VPN example

Site-to-site IPv6 over IPv6 VPN example

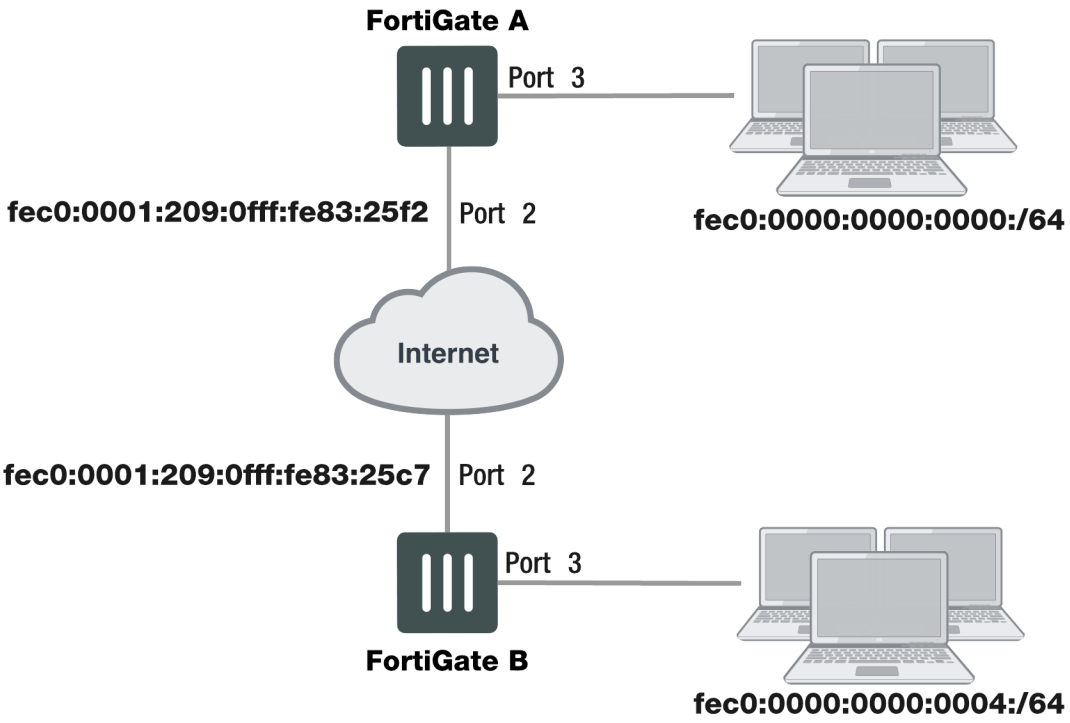
In this example, computers on IPv6-addressed private networks communicate securely over public IPv6 infrastructure.

By default IPv6 configurations to not appear on the Web-based Manager. You need to enable the feature first.

To enable IPv6

1. Go to **System > Feature Visibility**.
2. Enable **IPv6**.
3. Select **Apply**.

Example IPv6-over-IPv6 VPN topology



Configure FortiGate A interfaces

Port 2 connects to the public network and port 3 connects to the local network.

```
config system interface
   edit port2
      config ipv6
         set ip6-address fec0::0001:209:0fff:fe83:25f2/64
      end
   next
   edit port3
      config ipv6
         set ip6-address fec0::0000:209:0fff:fe83:25f3/64
      end
   next
end
```

Configure FortiGate A IPsec settings

The Phase 1 configuration creates a virtual IPsec interface on port 2 and sets the remote gateway to the public IP address FortiGate B. This configuration is the same as for an IPv4 route-based VPN, except that `ip-version` is set to `6` and the `remote-gw6` keyword is used to specify an IPv6 remote gateway address.

```
config vpn ipsec phase1-interface
   edit toB
      set ip-version 6
      set interface port2
      set remote-gw6 fec0:0000:0000:0003:209:0fff:fe83:25c7
      set dpd [disable | on-idle | on-demand]
      set psksecret maryhadalittlelamb
      set proposal 3des-md5 3des-sha1
   end
```

By default, Phase 2 selectors are set to accept all subnet addresses for source and destination. The default setting for `src-addr-type` and `dst-addr-type` is `subnet`. The IPv6 equivalent is `subnet6`. The default subnet addresses are 0.0.0.0/0 for IPv4, `::/0` for IPv6.

```
config vpn ipsec phase2-interface
   edit toB2
      set phase1name toB
      set proposal 3des-md5 3des-sha1
      set pfs enable
      set replay enable
      set src-addr-type subnet6
      set dst-addr-type subnet6
   end
```

Configure FortiGate A security policies

Security policies are required to allow traffic between port3 and the IPsec interface toB in each direction. The address `all6` must be defined using the `firewall address6` command as `::/0`.

```
config firewall policy6
   edit 1
      set srcintf port3
      set dstintf toB
      set srcaddr all6
```

```
      set dstaddr all6
      set action accept
      set service ANY
      set schedule always
   next
   edit 2
      set srcintf toB
      set dstintf port3
      set srcaddr all6
      set dstaddr all6
      set action accept
      set service ANY
      set schedule always
   end
```

Configure FortiGate A routing

This simple example requires just two static routes. Traffic to the protected network behind FortiGate B is routed via the virtual IPsec interface toB. A default route sends all IPv6 traffic out on port2.

```
config router static6
   edit 1
      set device port2
      set dst 0::/0
   next
   edit 2
      set device toB
      set dst fec0:0000:0000:0004::/64
   end
```

Configure FortiGate B

The configuration of FortiGate B is very similar to that of FortiGate A. A virtual IPsec interface toA is configured on port2 and its remote gateway is the public IP address of FortiGate A. Security policies enable traffic to pass between the private network and the IPsec interface. Routing ensures traffic for the private network behind FortiGate A goes through the VPN and that all IPv6 packets are routed to the public network.

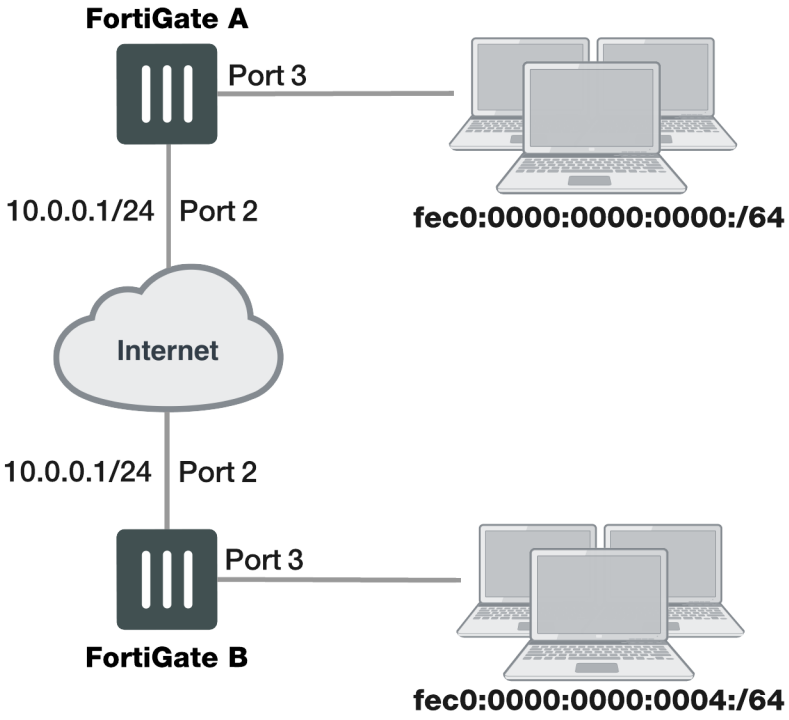
```
config system interface
   edit port2
      config ipv6
         set ip6-address fec0::0003:209:0fff:fe83:25c7/64
      end
   next
   edit port3
      config ipv6
         set ip6-address fec0::0004:209:0fff:fe83:2569/64
      end
   end
config vpn ipsec phase1-interface
   edit toA
      set ip-version 6
      set interface port2
      set remote-gw6 fec0:0000:0000:0001:209:0fff:fe83:25f2
      set dpd [disable | on-idle | on-demand]
      set psksecret maryhadalittlelamb
      set proposal 3des-md5 3des-sha1
   end
config vpn ipsec phase2-interface
```

```
    edit toA2
      set phase1name toA
      set proposal 3des-md5 3des-sha1
      set pfs enable
      set replay enable
      set src-addr-type subnet6
      set dst-addr-type subnet6
    end
  config firewall policy6
    edit 1
      set srcintf port3
      set dstintf toA
      set srcaddr all6
      set dstaddr all6
      set action accept
      set service ANY
      set schedule always
    next
    edit 2
      set srcintf toA
      set dstintf port3
      set srcaddr all6
      set dstaddr all6
      set action accept
      set service ANY
      set schedule always
    end
  config router static6
    edit 1
      set device port2
      set dst 0::/0
    next
    edit 2
      set device toA
      set dst fec0:0000:0000:0000::/64
  end
```

Site-to-site IPv6 over IPv4 VPN example

In this example, IPv6-addressed private networks communicate securely over IPv4 public infrastructure.

Example IPv6-over-IPv4 VPN topology



Configure FortiGate A interfaces

Port 2 connects to the IPv4 public network and port 3 connects to the IPv6 LAN.

```
config system interface
   edit port2
      set 10.0.0.1/24
      next
      edit port3
      config ipv6
         set ip6-address fec0::0001:209:0fff:fe83:25f3/64
      end
```

Configure FortiGate A IPsec settings

The Phase 1 configuration uses IPv4 addressing.

```
config vpn ipsec phase1-interface
   edit toB
      set interface port2
      set remote-gw 10.0.1.1
      set dpd [disable | on-idle | on-demand]
      set psksecret maryhadalittlelamb
      set proposal 3des-md5 3des-sha1
   end
```

The Phase 2 configuration uses IPv6 selectors. By default, Phase 2 selectors are set to accept all subnet addresses for source and destination. The default setting for `src-addr-type` and `dst-addr-type` is `subnet`. The IPv6 equivalent is `subnet6`. The default subnet addresses are 0.0.0.0/0 for IPv4, `::/0` for IPv6.

```
config vpn ipsec phase2-interface
   edit toB2
      set phase1name toB
      set proposal 3des-md5 3des-sha1
      set pfs enable
      set replay enable
      set src-addr-type subnet6
      set dst-addr-type subnet6
   end
```

Configure FortiGate A security policies

IPv6 security policies are required to allow traffic between port3 and the IPsec interface toB in each direction. Define the address `all6` using the `firewall address6` command as `::/0`.

```
config firewall policy6
   edit 1
      set srcintf port3
      set dstintf toB
      set srcaddr all6
      set dstaddr all6
      set action accept
      set service ANY
      set schedule always
   next
   edit 2
      set srcintf toB
      set dstintf port3
      set srcaddr all6
      set dstaddr all6
      set action accept
      set service ANY
      set schedule always
   end
```

Configure FortiGate A routing

This simple example requires just two static routes. Traffic to the protected network behind FortiGate B is routed via the virtual IPsec interface toB using an IPv6 static route. A default route sends all IPv4 traffic, including the IPv4 IPsec packets, out on port2.

```
config router static6
      edit 1
         set device toB
         set dst fec0:0000:0000:0004::/64
      end
   config router static
      edit 1
         set device port2
         set dst 0.0.0.0/0
         set gateway 10.0.0.254
      end
```

Configure FortiGate B

The configuration of FortiGate B is very similar to that of FortiGate A. A virtual IPsec interface toA is configured on port2 and its remote gateway is the IPv4 public IP address of FortiGate A. The IPsec Phase 2 configuration has IPv6 selectors.

IPv6 security policies enable traffic to pass between the private network and the IPsec interface. An IPv6 static route ensures traffic for the private network behind FortiGate A goes through the VPN and an IPv4 static route ensures that all IPv4 packets are routed to the public network.

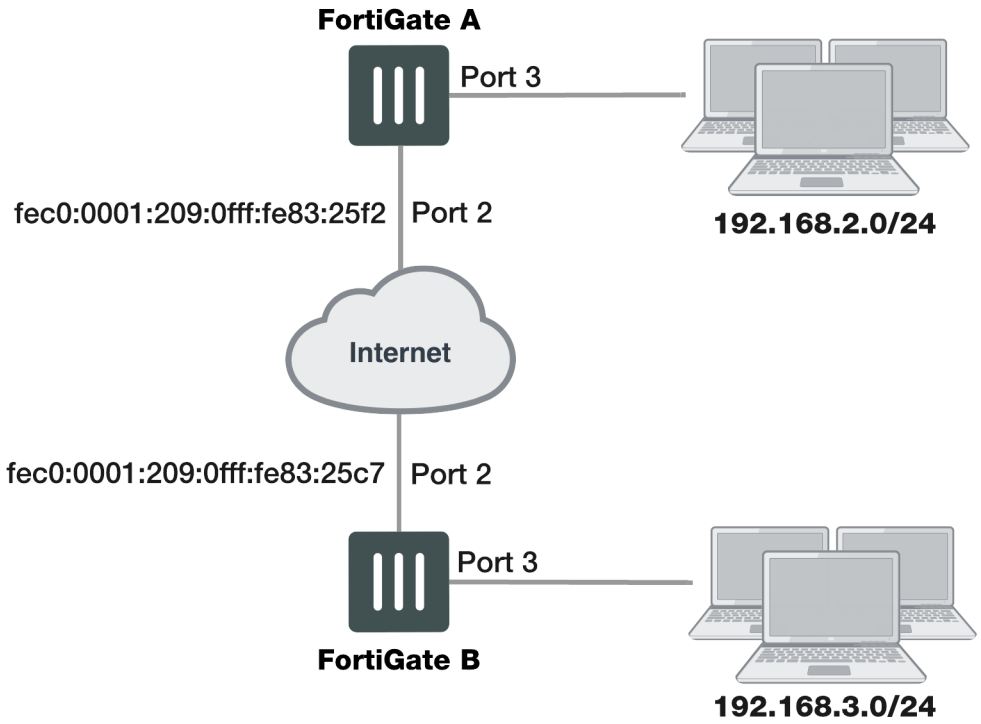
```
config system interface
   edit port2
      set 10.0.1.1/24
   next
   edit port3
      config ipv6
         set ip6-address fec0::0004:209:0fff:fe83:2569/64
      end
config vpn ipsec phase1-interface
   edit toA
      set interface port2
      set remote-gw 10.0.0.1
      set dpd [disable | on-idle | on-demand]
      set psksecret maryhadalittlelamb
      set proposal 3des-md5 3des-sha1
   end
config vpn ipsec phase2-interface
   edit toA2
      set phase1name toA
      set proposal 3des-md5 3des-sha1
      set pfs enable
      set replay enable
      set src-addr-type subnet6
      set dst-addr-type subnet6
   end
config firewall policy6
   edit 1
      set srcintf port3
      set dstintf toA
      set srcaddr all6
      set dstaddr all6
      set action accept
      set service ANY
      set schedule always
   next
   edit 2
      set srcintf toA
      set dstintf port3
      set srcaddr all6
      set dstaddr all6
      set action accept
      set service ANY
      set schedule always
   end
config router static6
   edit 1
```

```
      set device toA
      set dst fec0:0000:0000:0000::/64
   end
config router static
   edit 1
      set device port2
      set gateway 10.0.1.254
   end
```

Site-to-site IPv4 over IPv6 VPN example

In this example, two private networks with IPv4 addressing communicate securely over IPv6 infrastructure.

Example IPv4-over-IPv6 VPN topology



Configure FortiGate A interfaces

Port 2 connects to the IPv6 public network and port 3 connects to the IPv4 LAN.

```
config system interface
   edit port2
      config ipv6
         set ip6-address fec0::0001:209:0fff:fe83:25f2/64
      end
   next
   edit port3
      set 192.168.2.1/24
   end
```

Configure FortiGate A IPsec settings

The Phase 1 configuration is the same as in the IPv6 over IPv6 example.

```
config vpn ipsec phase1-interface
   edit toB
      set ip-version 6
      set interface port2
      set remote-gw6 fec0:0000:0000:0003:209:0fff:fe83:25c7
      set dpd [disable | on-idle | on-demand]
      set psksecret maryhadalittlelamb
      set proposal 3des-md5 3des-sha1
   end
```

The Phase 2 configuration is the same as you would use for an IPv4 VPN. By default, Phase 2 selectors are set to accept all subnet addresses for source and destination.

```
config vpn ipsec phase2-interface
   edit toB2
      set phase1name toB
      set proposal 3des-md5 3des-sha1
      set pfs enable
      set replay enable
   end
```

Configure FortiGate A security policies

Security policies are required to allow traffic between port3 and the IPsec interface toB in each direction. These are IPv4 security policies.

```
config firewall policy
   edit 1
      set srcintf port3
      set dstintf toB
      set srcaddr all
      set dstaddr all
      set action accept
      set service ANY
      set schedule always
   next
   edit 2
      set srcintf toB
      set dstintf port3
      set srcaddr all
      set dstaddr all
      set action accept
      set service ANY
      set schedule always
   end
```

Configure FortiGate A routing

This simple example requires just two static routes. Traffic to the protected network behind FortiGate B is routed via the virtual IPsec interface toB using an IPv4 static route. A default route sends all IPv6 traffic, including the IPv6 IPsec packets, out on port2.

```
config router static6
```

```
   edit 1
      set device port2
      set dst 0::/0
   next
   edit 2
      set device toB
      set dst 192.168.3.0/24
   end
```

Configure FortiGate B

The configuration of FortiGate B is very similar to that of FortiGate A. A virtual IPsec interface toA is configured on port2 and its remote gateway is the public IP address of FortiGate A. The IPsec Phase 2 configuration has IPv4 selectors.

IPv4 security policies enable traffic to pass between the private network and the IPsec interface. An IPv4 static route ensures traffic for the private network behind FortiGate A goes through the VPN and an IPv6 static route ensures that all IPv6 packets are routed to the public network.

```
config system interface
   edit port2
      config ipv6
         set ip6-address fec0::0003:fe83:25c7/64
      end
   next
   edit port3
      set 192.168.3.1/24
   end
config vpn ipsec phase1-interface
   edit toA
      set ip-version 6
      set interface port2
      set remote-gw6 fec0:0000:0000:0001:209:0fff:fe83:25f2
      set dpd [disable | on-idle | on-demand]
      set psksecret maryhadalittlelamb
      set proposal 3des-md5 3des-sha1
   end
config vpn ipsec phase2-interface
   edit toA2
      set phase1name toA
      set proposal 3des-md5 3des-sha1
      set pfs enable
      set replay enable
   end
config firewall policy
   edit 1
      set srcintf port3
      set dstintf toA
      set srcaddr all
      set dstaddr all
      set action accept
      set service ANY
      set schedule always
   next
   edit 2
      set srcintf toA
      set dstintf port3
```

```
      set srcaddr all
      set dstaddr all
      set action accept
      set service ANY
      set schedule always
   end
config router static6
   edit 1
      set device port2
      set dst 0::/0
   next
   edit 2
      set device toA
      set dst 192.168.2.0/24
   end
```

L2TP and IPsec (Microsoft VPN)

This section describes how to set up a VPN that is compatible with the Microsoft Windows native VPN, which is Layer 2 Tunneling Protocol (L2TP) with IPsec encryption.

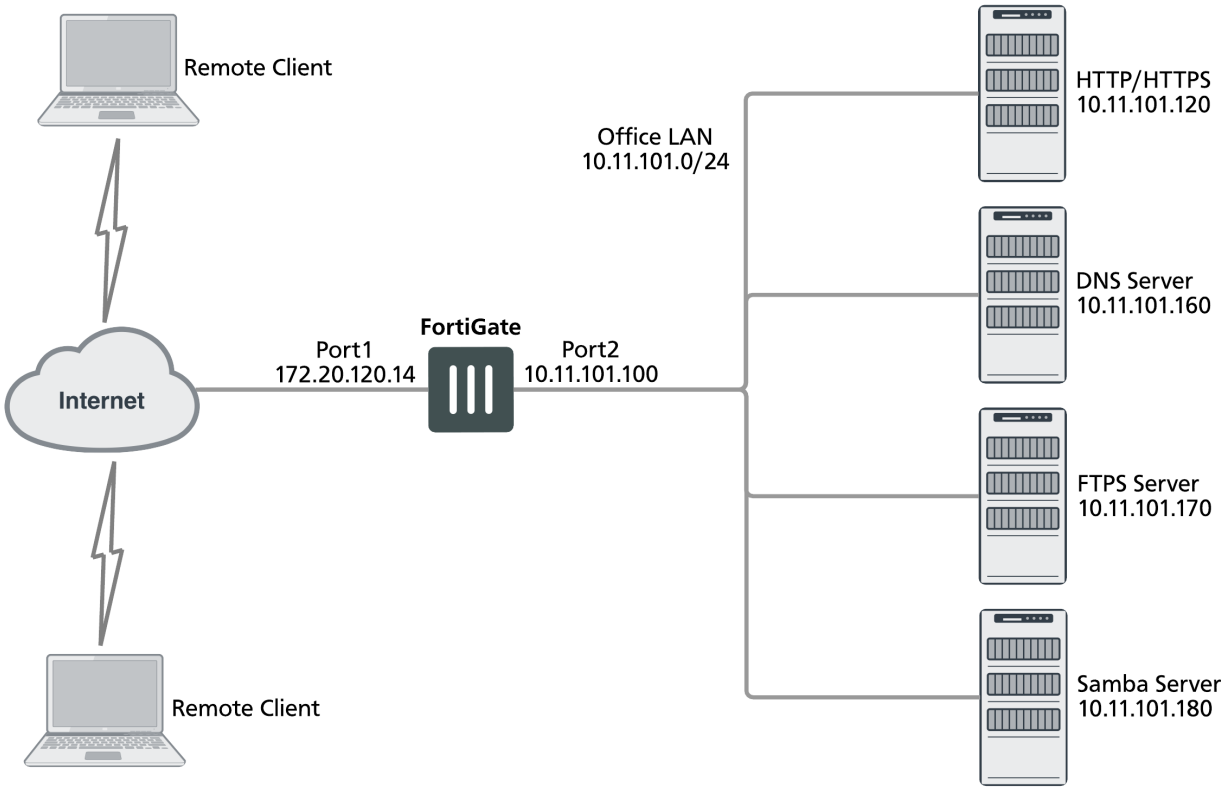
The following topics are included in this section:

For troubleshooting information, refer to Troubleshooting L2TP and IPsec.

Overview

The topology of a VPN for Microsoft Windows dialup clients is very similar to the topology for FortiClient Endpoint Security clients.

Example FortiGate VPN configuration with Microsoft clients



For users, the difference is that instead of installing and using the FortiClient application, they configure a network connection using the software built into the Microsoft Windows operating system. Starting in FortiOS 4.0 MR2, you can configure a FortiGate unit to work with unmodified Microsoft VPN client software.

Layer 2 Tunneling Protocol (L2TP)

L2TP is a tunneling protocol published in 1999 that is used with VPNs, as the name suggests. Microsoft Windows operating system has a built-in L2TP client starting since Windows 2000. Mac OS X 10.3 system and higher also have a built-in client.

L2TP provides no encryption and used UDP port 1701. IPsec is used to secure L2TP packets. The initiator of the L2TP tunnel is called the L2TP Access Concentrator (LAC).

L2TP and IPsec is supported for native Windows XP, Windows Vista and Mac OSX native VPN clients. However, in Mac OSX (OSX 10.6.3, including patch releases) the L2TP feature does not work properly on the Mac OS side.

Assumptions

The following assumptions have been made for this example:

- L2TP protocol traffic is allowed through network firewalls (TCP and UDP port 1701)
- User has Microsoft Windows 2000 or higher — a Windows version that supports L2TP

Configuration overview

The following section consists of configuring the FortiGate unit and configuring the Windows PC.

Configuring the FortiGate unit

To configure the FortiGate unit, you must:

- Configure LT2P users and firewall user group.
- Configure the L2TP VPN, including the IP address range it assigns to clients.
- Configure an IPsec VPN with encryption and authentication settings that match the Microsoft VPN client.
- Configure security policies.

Configuring LT2P users and firewall user group

Remote users must be authenticated before they can request services and/or access network resources through the VPN. The authentication process can use a password defined on the FortiGate unit or an established external authentication mechanism such as RADIUS or LDAP.

Creating user accounts

You need to create user accounts and then add these users to a firewall user group to be used for L2TP authentication. The Microsoft VPN client can automatically send the user's Window network logon credentials. You might want to use these for their L2TP user name and password.

Creating a user account - web-based manager

1. Go to **User & Device > User Definition**and select **Create New**.
2. Enter the **User Name**.
3. Do one of the following:
   - Select **Password** and enter the user's assigned password.
   - Select **Match user on LDAP server**, **Match user on RADIUS server**, or **Match user onTACACS+ server** and select the authentication server from the list. The authentication server must be already configured on the FortiGate unit.
4. Select **OK**.

Creating a user account - CLI

To create a user account called `user1` with the password `123_user`, enter:

```
config user local
   edit user1
      set type password
      set passwd "123_user"
      set status enable
   end
```

Creating a user group

When clients connect using the L2TP-over-IPsec VPN, the FortiGate unit checks their credentials against the user group you specify for L2TP authentication. You need to create a firewall user group to use for this purpose.

Creating a user group - web-based manager

1. Go to **User & Device > User Groups**, select **Create New**, and enter the following:

Name	Type or edit the user group name (for example, `L2TP_group).`
Type	Select **Firewall**.
Available Users/Groups	The list of Local users, RADIUS servers, LDAP servers, TACACS+ servers, or PKI users that can be added to the user group. To add a member to this list, select the name and then select the right arrow button.
Members	The list of Local users, RADIUS servers, LDAP servers, TACACS+ servers, or PKI users that belong to the user group. To remove a member, select the name and then select the left arrow button.

2. Select **OK**.

Creating a user group - CLI

To create the user group L2TP_group and add members User_1, User_2, and User_3, enter:

```
config user group
   edit L2TP_group
      set group-type firewall
      set member User_1 User_2 User_3
```

```
end
```

Configuring L2TP

You can only configure L2TP settings in the CLI. As well as enabling L2TP, you set the range of IP address values that are assigned to L2TP clients and specify the user group that can access the VPN. For example, to allow access to users in the L2TP_group and assign them addresses in the range 192.168.0.50 to 192.168.0.59, enter:

```
config vpn l2tp
   set sip 192.168.0.50
   set eip 192.168.0.59
   set status enable
   set usrgrp "L2TP_group"
end
```

One of the security policies for the L2TP over IPsec VPN uses the client address range, so you need also need to create a firewall address for that range. For example,

```
config firewall address
   edit L2TPclients
      set type iprange
      set start-ip 192.168.0.50
      set end-ip 192.168.0.59
end
```

Alternatively, you could define this range in the web-based manager.

Configuring IPsec

The Microsoft VPN client uses IPsec for encryption. The configuration needed on the FortiGate unit is the same as for any other IPsec VPN with the following exceptions.

- Transport mode is used instead of tunnel mode.
- The encryption and authentication proposals must be compatible with the Microsoft client.

Whether Transport mode is *required* depends on the configuration of the peer device (typically an old Windows device, since newer versions of Windows don't require IPsec and L2TP—they can run IPsec natively).

When configuring L2TP, do not name the VPN "L2TP" as that will result in a conflict.

L2TP over IPsec is supported on the FortiGate unit for both policy-based and route-based configurations, but the following example is policy-based.

Configuring Phase 1 - web-based manager

1. Go to **VPN > IPsec Tunnels** and create the new custom tunnel or edit an existing tunnel.
2. Edit the **Phase 1 Proposal** (if it is not available, you may need to click the **Convert to Custom Tunnel** button).

Name	Enter a name for this VPN, dialup_p1 for example.
Remote Gateway	**Dialup User**
Local Interface	Select the network interface that connects to the Internet. For example, port1.
Mode	**Main (ID protection)**
Authentication Method	**Preshared Key**
Pre-shared Key	Enter the preshared key. This key must also be entered in the Microsoft VPN client.
Advanced	Select **Advanced** to enter the following information.
Phase 1 Proposal	Enter the following Encryption/Authentication pairs: AES256-MD5, 3DES-SHA1, AES192-SHA1
Diffie-Hellman Group	2
NAT Traversal	Enable
Dead Peer Detection	Enable

Configuring Phase 1 - CLI

To create a Phase 1 configuration called dialup_p1 on a FortiGate unit that has port1 connected to the Internet, you would enter:

```
config vpn ipsec phase1
   edit dialup_p1
      set type dynamic
      set interface port1
      set mode main
      set psksecret ********
      set proposal aes256-md5 3des-sha1 aes192-sha1
      set dhgrp 2
      set nattraversal enable
      set dpd [disable | on-idle | on-demand]
end
```

It is worth noting here that the command `config vpn ipsec phase1` is used rather than `config vpn ipsec phase1-interface` because this configuration is policy-based and not route-based.

Configuring Phase 2 - web-based manager

1. Open the **Phase 2 Selectors** panel.
2. Enter the following information and then select **OK**.

Phase 2 Proposal	Enter the following Encryption/Authentication pairs: AES256-MD5, 3DES-SHA1, AES192-SHA1
Enable replay detection	Enable
Enable perfect forward secrecy (PFS)	Disable
Keylife	3600 seconds

3. Make this a transport-mode VPN. You must use the CLI to do this. If your Phase 2 name is dialup_p2, you would enter:

```
config vpn ipsec phase2
   edit dialup_p2
      set encapsulation transport-mode
   end
```

Configuring Phase 2 - CLI

To configure a Phase 2 to work with your phase_1 configuration, you would enter:

```
config vpn ipsec phase2
   edit dialup_p2
      set phase1name dialup_p1
      set proposal aes256-md5 3des-sha1 aes192-sha1
      set replay enable
      set pfs disable
      set keylifeseconds 3600
      set encapsulation transport-mode
   end
```

Once again, note here that the command `config vpn ipsec phase2` is used rather than `config vpn ipsec phase2-interface` because this configuration is policy-based and not route-based.

Configuring security policies

The security policies required for L2TP over IPsec VPN are:

- An IPsec policy, as you would create for any policy-based IPsec VPN
- A regular ACCEPT policy to allow traffic from the L2TP clients to access the protected network

Configuring the IPsec security policy - web-based manager

1. Go to **System > Feature Visibility** and enable **Policy-based IPsec VPN**.
2. Go to **Policy & Objects > IPv4 Policy** and select **Create New**.
3. Set the **Action** to **IPsec** and enter the following information:

Incoming Interface	Select the interface that connects to the private network behind this FortiGate unit.

Source Address	All
Outgoing Interface	Select the FortiGate unit's public interface.
Destination Address	All
VPN Tunnel	Select **Use Existing** and select the name of the Phase 1 configuration that you created. For example, dialup_p1. See Configuring IPsec on page 181.
Allow traffic to be initiated from the remote site	enable

4. Select **OK**.

Configuring the IPsec security policy - CLI

If your VPN tunnel (Phase 1) is called dialup_p1, your protected network is on port2, and your public interface is port1, you would enter:

```
config firewall policy
   edit 0
      set srcintf port2
      set dstintf port1
      set srcaddr all
      set dstaddr all
      set action ipsec
      set schedule always
      set service all
      set inbound enable
      set vpntunnel dialup_p1
   end
```

Configuring the ACCEPT security policy - web-based manager

1. Go to **Policy & Objects > IPv4 Policy** and select **Create New**.
2. Leave the **Policy Type** as **Firewall** and leave the **Policy Subtype** as **Address**.
3. Enter the following information and select **OK**:

Incoming Interface	Select the FortiGate unit's public interface.
Source Address	Select the firewall address that you defined for the L2TP clients.
Outgoing Interface	Select the interface that connects to the private network behind this FortiGate unit.
Destination Address	All
Action	ACCEPT

Configuring the ACCEPT security policy - CLI

If your public interface is port1, your protected network is on port2, and L2TPclients is the address range that L2TP clients use, you would enter:

```
config firewall policy
```

```
edit 1
   set srcintf port1
   set dstintf port2
   set srcaddr L2TPclients
   set dstaddr all
   set action accept
   set schedule always
   set service all
end
```

Configuring the Windows PC

Configuration of the Windows PC for a VPN connection to the FortiGate unit consists of the following:

1. In Network Connections, configure a Virtual Private Network connection to the FortiGate unit.
2. Ensure that the IPSEC service is running.
3. Ensure that IPsec has not been disabled for the VPN client. It may have been disabled to make the Microsoft VPN compatible with an earlier version of FortiOS.

The instructions in this section are based on Windows XP. Other versions of Windows may vary slightly.

Configuring the network connection

1. Open **Network Connections**.
   This is available through the Control Panel.
2. Double-click **New Connection Wizard** and **Select Next**.
3. Select **Connect to the network at my workplace**.
4. Select **Next**.
5. Select **Virtual Private Network connection** and select **Next**.
6. In the **Company Name** field, enter a name for the connection and select **Next**.
7. Select **Do not dial the initial connection** and then select **Next**.
8. Enter the public IP address or FQDN of the FortiGate unit and select **Next**.
9. Optionally, select **Add a shortcut to this connection to my desktop**.
10. Select **Finish**.
    The **Connect** dialog opens on the desktop.
11. Select **Properties** and then select the **Security** tab.
12. Select **IPsec Settings**.
13. Select **Use pre-shared key for authentication**, enter the preshared key that you configured for your VPN, and select **OK**.
14. Select **OK**.

Checking that the IPsec service is running

1. Open **Administrative Tools** through the Control Panel.
2. Double-click **Services**.
3. Look for IPSEC Services. Confirm that the **Startup Type** is **Automatic** and **Status** is set to **Started**. If needed, double-click **IPsec Services** to change these settings.

Checking that IPsec has not been disabled

1. Select **Start > Run**.
2. Enter regedit and select **OK**.
3. Find the Registry key HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\RasMan\Parameters
4. If there is a ProhibitIPsec value, it must be set to 0.

Enforcing IPsec in L2TP configuration

An `enforce-ipsec` option is available in L2TP configuration to force the FortiGate L2TP server to accept only IPsec encrypted connections.

Syntax

```
config vpn l2tp
   set eip 50.0.0.100
   set sip 50.0.0.1
   set status enable
   set enforce-ipsec-interface {disable | enable}      (default = disable)
   set usrgrp <group_name>
end
```

GRE over IPsec (Cisco VPN)

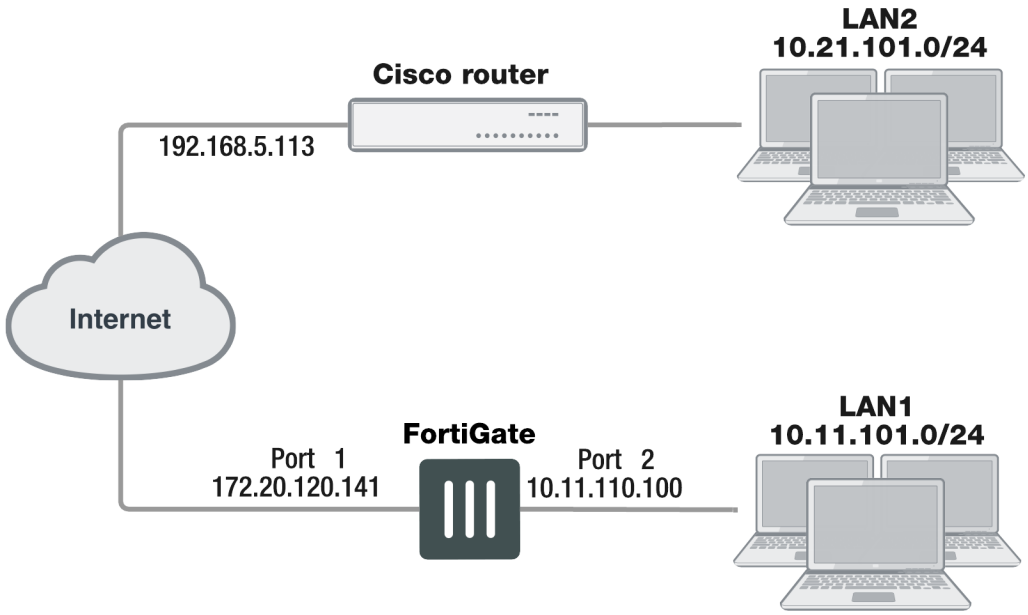
This section describes how to configure a FortiGate VPN that is compatible with Cisco-style VPNs that use GRE in an IPsec tunnel.

The following topics are included in this section:

Cisco products that include VPN support often use Generic Routing Encapsulation (GRE) protocol tunnel over IPsec encryption. This chapter describes how to configure a FortiGate unit to work with this type of Cisco VPN.

Cisco VPNs can use either transport mode or tunnel mode IPsec. Before FortiOS 4.0 MR2, the FortiGate unit was compatible only with tunnel mode IPsec.

Example FortiGate to Cisco GRE-over-IPsec VPN



In this example, users on LAN1 are provided access to LAN2.

Configuration overview

The following section consists of configuring the FortiGate unit and configuring the Cisco router.

Configuring the FortiGate unit

There are several steps to the GRE-over-IPsec configuration:

- Enable overlapping subnets. This is needed because the IPsec and GRE tunnels will use the same addresses.
- Configure a route-based IPsec VPN on the external interface.
- Configure a GRE tunnel on the virtual IPsec interface. Set its local gateway and remote gateway addresses to match the local and remote gateways of the IPsec tunnel.
- Configure security policies to allow traffic to pass in both directions between the GRE virtual interface and the IPsec virtual interface.
- Configure security policies to allow traffic to pass in both directions between the protected network interface and the GRE virtual interface.
- Configure a static route to direct traffic destined for the network behind the Cisco router into the GRE-over-IPsec tunnel.

Enabling overlapping subnets

By default, each FortiGate unit network interface must be on a separate network. The configuration described in this chapter assigns an IPsec tunnel end point and the external interface to the same network. Enable subnet overlap as follows:

```
config system settings
   set allow-subnet-overlap enable
end
```

Configuring the IPsec VPN

A route-based VPN is required. It must use encryption and authentication algorithms compatible with the Cisco equipment to which it connects. In this chapter, preshared key authentication is shown.

Configuring the IPsec VPN - web-based manager

1. Define the Phase 1 configuration needed to establish a secure connection with the remote Cisco device. Enter these settings in particular:

Name	Enter a name to identify the VPN tunnel, tocisco for example. This is the name of the virtual IPsec interface. It appears in Phase 2 configurations, security policies and the VPN monitor.
Remote Gateway	Select **Static IP Address**.
IP Address	Enter the IP address of the Cisco device public interface. For example, 192.168.5.113.

Local Interface	Select the FortiGate unit's public interface. For example, 172.20.120.141.
Mode	Select **Main (ID Protection)**.
Authentication Method	**Preshared Key**
Pre-shared Key	Enter the preshared key. It must match the preshared key on the Cisco device.
Advanced	Select the Advanced button to see the following settings.
Phase 1 Proposal	**3DES-MD5** At least one proposal must match the settings on the Cisco unit.

For more information about these settings, see Phase 1 parameters on page 46.

2. Define the Phase 2 parameters needed to create a VPN tunnel with the remote peer. For compatibility with the Cisco router, Quick Mode Selectors must be entered, which includes specifying protocol 47, the GRE protocol. Enter these settings in particular:

Phase 2 Proposal	**3DES-MD5** At least one proposal must match the settings on the Cisco unit.
Quick Mode Selector	
Source Address	Enter the GRE local tunnel end IP address. For example 172.20.120.141.
Source Port	0
Destination Address	Enter the GRE remote tunnel end IP address. For example 192.168.5.113.
Destination Port	0
Protocol	47

For more information about these settings, see Phase 2 parameters on page 66.

3. If the Cisco device is configured to use transport mode IPsec, you need to use transport mode on the FortiGate VPN. You can configure this only in the CLI. In your Phase 2 configuration, set `encapsulation` to `transport-mode` as follows:

```
config vpn phase2-interface
   edit to_cisco_p2
      set encapsulation transport-mode
   end
```

Configuring the IPsec VPN - CLI

```
config vpn ipsec phase1-interface
   edit tocisco
```

```
      set interface port1
      set proposal 3des-sha1 aes128-sha1
      set remote-gw 192.168.5.113
      set psksecret xxxxxxxxxxxxxxx
   end
config vpn ipsec phase2-interface
   edit tocisco_p2
      set phase1name "tocisco"
      set proposal 3des-md5
      set encapsulation tunnel-mode // if tunnel mode
      set encapsulation transport-mode // if transport mode
      set protocol 47
      set src-addr-type ip
      set dst-start-ip 192.168.5.113
      set src-start-ip 172.20.120.141
   end
```

Adding IPsec tunnel end addresses

The Cisco configuration requires an address for its end of the IPsec tunnel. The addresses are set to match the GRE gateway addresses. Use the CLI to set the addresses, like this:

```
config system interface
   edit tocisco
      set ip 172.20.120.141 255.255.255.255
      set remote-ip 192.168.5.113
   end
```

Configuring the GRE tunnel

The GRE tunnel runs between the virtual IPsec public interface on the FortiGate unit and the Cisco router. You must use the CLI to configure a GRE tunnel. In the example, you would enter:

```
config system gre-tunnel
   edit gre1
      set interface tocisco
      set local-gw 172.20.120.141
      set remote-gw 192.168.5.113
end
```

`interface` is the virtual IPsec interface, `local-gw` is the FortiGate unit public IP address, and `remote-gw` is the remote Cisco device public IP address

Adding GRE tunnel end addresses

You will also need to add tunnel end addresses. The Cisco router configuration requires an address for its end of the GRE tunnel. Using the CLI, enter tunnel end addresses that are not used elsewhere on the FortiGate unit, like this:

```
config system interface
   edit gre1
      set ip 10.0.1.1 255.255.255.255
      set remote-ip 10.0.1.2
   end
```

Configuring security policies

Two sets of security policies are required:

- Policies to allow traffic to pass in both directions between the GRE virtual interface and the IPsec virtual interface.
- Policies to allow traffic to pass in both directions between the protected network interface and the GRE virtual interface.

Configuring security policies - web-based manager

1. Define an ACCEPT firewall security policy to permit communications between the protected network and the GRE tunnel:

Incoming Interface	Select the interface that connects to the private network behind this FortiGate unit.
Source Address	**All**
Outgoing Interface	Select the GRE tunnel virtual interface you configured.
Destination Address	**All**
Action	**ACCEPT**
Enable NAT	Disable

2. To permit the remote client to initiate communication, you need to define a firewall address security policy for communication in that direction:

Incoming Interface	Select the GRE tunnel virtual interface you configured.
Source Address	**All**
Outgoing Interface	Select the interface that connects to the private network behind this FortiGate unit.
Destination Address	**All**
Action	**ACCEPT**
Enable NAT	Disable

3. Define a pair of ACCEPT firewall address security policies to permit traffic to flow between the GRE virtual interface and the IPsec virtual interface:

Incoming Interface	Select the GRE virtual interface. See Configuring the GRE tunnel on page 190.
Source Address	**All**
Outgoing Interface	Select the virtual IPsec interface you created. See Configuring the IPsec VPN on page 188.
Destination Address	**All**
Action	**ACCEPT**
Enable NAT	Disable

Incoming Interface	Select the virtual IPsec interface you created. See Configuring the IPsec VPN on page 188.
Source Address	**All**
Outgoing Interface	Select the GRE virtual interface.See Configuring the GRE tunnel on page 190.
Destination Address	**All**
Action	**ACCEPT**
Enable NAT	Disable

Configuring security policies - CLI

```
config firewall policy
   edit 1 // LAN to GRE tunnel
      set srcintf port2
      set dstintf gre1
      set srcaddr all
      set dstaddr all
      set action accept
      set schedule always
      set service ANY
   next
   edit 2 // GRE tunnel to LAN
      set srcintf gre1
      set dstintf port2
      set srcaddr all
      set dstaddr all
      set action accept
      set schedule always
      set service ANY
   next
   edit 3 // GRE tunnel to IPsec interface
      set srcintf "gre1"
      set dstintf "tocisco"
      set srcaddr "all"
      set dstaddr "all"
      set action accept
      set schedule "always"
      set service "ANY"
   next
   edit 4 // IPsec interface to GRE tunnel
      set srcintf "tocisco"
      set dstintf "gre1"
      set srcaddr "all"
      set dstaddr "all"
      set action accept
      set schedule "always"
      set service "ANY"
   end
```

Configuring routing

Traffic destined for the network behind the Cisco router must be routed to the GRE tunnel. To do this, create a static route

1. Go to **Network > Static Routes** and select **Create New**.
2. Enter the following information and select **OK**.

Destination IP/Mask	Enter the IP address and netmask for the network behind the Cisco router. For example 10.21.101.0 255.255.255.0.
Device	Select the GRE virtual interface.
Distance (Advanced)	Leave setting at default value.

In the CLI, using the example values, you would enter

```
config router static
   edit 0
      set device gre1
      set dst 10.21.101.0 255.255.255.0
   end
```

Changing GRE over GRE tunnel interface attributes

Administrators can change GRE over GRE tunnel attributes, such as assigning an IP address for a specific configuration application, even if the child interface is not an IPsec tunnel interface.

IPv6 support for GRE tunnels

Support is provided for GRE tunnel termination using IPv6 addresses on both ends of the tunnel (similar to IPv4 functionality).

Syntax

```
config system gre-tunnel
   edit <name>
      set ip-version 6
      set remote-gw6 11:1:1:1::1
      set local-gw6 11:1:1:1::2
      ...
   next
end
```

Configuring the Cisco router

Using Cisco IOS, you would configure the Cisco router as follows, using the addresses from the example:

```
config ter
crypto ipsec transform-set myset esp-3des esp-md5-hmac
no mode
exit
no ip access-list extended tunnel
```

```
ip access-list extended tunnel
permit gre host 192.168.5.113 host 172.20.120.141
exit
interface Tunnel1
ip address 10.0.1.2 255.255.255.0
tunnel source 192.168.5.113
tunnel destination 172.20.120.141
!
ip route 10.11.101.0 255.255.255.0 Tunnel1
end
clea crypto sa
clea crypto isakmp
```

For transport mode, change `no mode` to `mode transport`.

This is only the portion of the Cisco router configuration that applies to the GRE-over-IPsec tunnel. For more information, refer to the Cisco documentation.

Keep-alive support for GRE

The FortiGate can send a GRE keep-alive response to a Cisco device to detect a GRE tunnel. If it fails, it will remove any routes over the GRE interface.

Syntax

```
config system gre-tunnel
   edit <id>
      set keepalive-interval <value: 0-32767>
      set keepalive-failtimes <value: 1-255>
   next
end
```

Protecting OSPF with IPsec

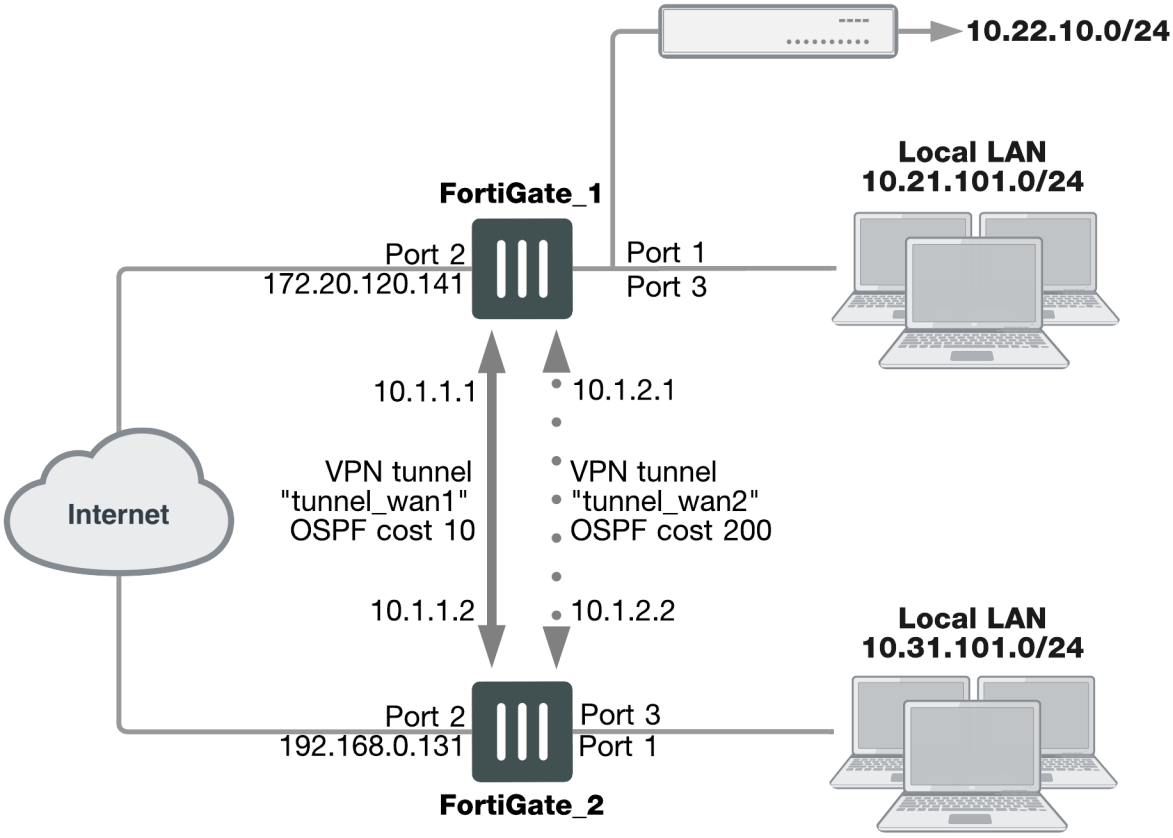
For enhanced security, OSPF dynamic routing can be carried over IPsec VPN links.

The following topics are included in this section:

Configuration overview

This chapter shows an example of OSPF routing conducted over an IPsec tunnel between two FortiGate units. The network shown below is a single OSPF area. FortiGate_1 is an Area border router that advertises a static route to 10.22.10.0/24 in OSPF. FortiGate_2 advertises its local LAN as an OSPF internal route.

OSPF over an IPsec VPN tunnel



The section Configuration overview describes the configuration with only one IPsec VPN tunnel, tunnel_wan1. Then, the section Configuration overview describes how you can add a second tunnel to provide a redundant backup path. This is shown above as VPN tunnel “tunnel_wan2”.

Only the parts of the configuration concerned with creating the IPsec tunnel and integrating it into the OSPF network are described. It is assumed that security policies are already in place to allow traffic to flow between the interfaces on each FortiGate unit.

OSPF over IPsec configuration

There are several steps to the OSPF-over-IPsec configuration:

- Configure a route-based IPsec VPN on an external interface. It will connect to a corresponding interface on the other FortiGate unit. Define the two tunnel-end addresses.
- Configure a static route to the other FortiGate unit.
- Configure the tunnel network as part of the OSPF network and define the virtual IPsec interface as an OSPF interface.

This section describes the configuration with only one VPN, tunnel_wan1. The other VPN is added in the section Configuration overview on page 196.

Configuring the IPsec VPN

A route-based VPN is required. In this chapter, preshared key authentication is shown. Certificate authentication is also possible. Both FortiGate units need this configuration.

Configuring Phase 1

1. Define the Phase 1 configuration needed to establish a secure connection with the other FortiGate unit. For more information, see Phase 1 parameters on page 46.

   Enter these settings in particular:

Name	Enter a name to identify the VPN tunnel, tunnel_wan1 for example. This becomes the name of the virtual IPsec interface.
Remote Gateway	Select **Static IP Address**.
IP Address	Enter the IP address of the other FortiGate unit's public (Port 2) interface.
Local Interface	Select this FortiGate unit's public (Port 2) interface.
Mode	Select **Main (ID Protection)**.
Authentication Method	**Preshared Key**
Pre-shared Key	Enter the preshared key. It must match the preshared key on the other FortiGate unit.
Advanced	Select **Advanced**.

Assigning the tunnel end IP addresses

1. Go to **Network > Interfaces**, select the virtual IPsec interface that you just created on Port 2 and select **Edit**.
2. In the **IP** and **Remote IP** fields, enter the following tunnel end addresses:

	FortiGate_1	FortiGate_2
IP	10.1.1.1	10.1.1.2
Remote_IP	10.1.1.2	10.1.1.1

These addresses are from a network that is not used for anything else.

Configuring Phase 2

1. Enter a name to identify this Phase 2 configuration, twan1_p2, for example.
2. Select the name of the Phase 1 configuration that you defined in Step "Configuration overview" on page 196, tunnel_wan1 for example.

Configuring static routing

You need to define the route for traffic leaving the external interface.

1. Go to **Network > Static Routes**, select **Create New**.
2. Enter the following information.

Destination IP/Mask	Leave as 0.0.0.0 0.0.0.0.
Device	Select the external interface.
Gateway	Enter the IP address of the next hop router.

Configuring OSPF

This section does not attempt to explain OSPF router configuration. It focusses on the integration of the IPsec tunnel into the OSPF network. This is accomplished by assigning the tunnel as an OSPF interface, creating an OSPF route to the other FortiGate unit.

This configuration uses loopback interfaces to ease OSPF troubleshooting. The OSPF router ID is set to the loopback interface address.The loopback interface ensures the router is always up. Even though technically the router ID doesn't have to match a valid IP address on the FortiGate unit, having an IP that matches the router ID makes troubleshooting a lot easier.

The two FortiGate units have slightly different configurations. FortiGate_1 is an AS border router that advertises its static default route. FortiGate_2 advertises its local LAN as an OSPF internal route.

Setting the router ID for each FortiGate unit to the lowest possible value is useful if you want the FortiGate units to be the designated router (DR) for their respective ASes. This is the router that broadcasts the updates for the AS.

Leaving the IP address on the OSPF interface at 0.0.0.0 indicates that all potential routes will be advertised, and it will not be limited to any specific subnet. For example if this IP address was 10.1.0.0, then only routes that match that subnet will be advertised through this interface in OSPF.

FortiGate_1 OSPF configuration

When configuring FortiGate_1 for OSPF, the loopback interface is created, and then you configure OSPF area networks and interfaces.

With the exception of creating the loopback interface, OSPF for this example can all be configured in either the web-based manager or CLI.

Creating the loopback interface

A loopback interface can be configured in the CLI only. For example, if the interface will have an IP address of 10.0.0.1, you would enter:

```
config system interface
   edit lback1
      set vdom root
         set ip 10.0.0.1 255.255.255.255
         set type loopback
   end
```

The loopback addresses and corresponding router IDs on the two FortiGate units must be different. For example, set the FortiGate 1 loopback to 10.0.0.1 and the FortiGate 2 loopback to 10.0.0.2.

Configuring OSPF area, networks, and interfaces - web-based manager

1. On FortiGate_1, go to **Network > OSPF**.
2. Enter the following information to define the router, area, and interface information.

Router ID	Enter `10.0.0.1`. Select **Apply** before entering the remaining information.
Advanced Options	
Redistribute	Select the **Connected** and **Static** check boxes. Use their default metric values.
Areas	Select **Create New**, enter the **Area** and **Type** and then select **OK**.
Area	0.0.0.0
Type	Regular
Interfaces	Enter a name for the OSPF interface, ospf_wan1 for example.
Name	
Interface	Select the virtual IPsec interface, tunnel_wan1.
IP	0.0.0.0

3. For **Networks**, select **Create New**.
4. Enter the **IP/Netmask** of `10.1.1.0/255.255.255.0` and an **Area** of `0.0.0.0`.
5. For **Networks**, select **Create New**.
6. Enter the **IP/Netmask** of `10.0.0.1/255.255.255.0` and an **Area** of `0.0.0.0`.
7. Select **Apply**.

Configuring OSPF area and interfaces - CLI

Your loopback interface is 10.0.0.1, your tunnel ends are on the 10.1.1.0/24 network, and your virtual IPsec interface is named `tunnel_wan1`. Enter the following CLI commands:

```
config router ospf
   set router-id 10.0.0.1
   config area
      edit 0.0.0.0
   end
   config network
      edit 4
         set prefix 10.1.1.0 255.255.255.0
      next
      edit 2
         set prefix 10.0.0.1 255.255.255.255
      end
   config ospf-interface
      edit ospf_wan1
         set cost 10
         set interface tunnel_wan1
         set network-type point-to-point
      end
   config redistribute connected
      set status enable
   end
   config redistribute static
      set status enable
   end
end
```

FortiGate_2 OSPF configuration

When configuring FortiGate_2 for OSPF, the loopback interface is created, and then you configure OSPF area networks and interfaces.

Configuring FortiGate_2 differs from FortiGate_1 in that three interfaces are defined instead of two. The third interface is the local LAN that will be advertised into OSPF.

With the exception of creating the loopback interface, OSPF for this example can all be configured in either the web-based manager or CLI.

Creating the loopback interface

A loopback interface can be configured in the CLI only. For example, if the interface will have an IP address of 10.0.0.2, you would enter:

```
config system interface
   edit lback1
      set vdom root
      set ip 10.0.0.2 255.255.255.255
      set type loopback
   end
```

The loopback addresses on the two FortiGate units must be different. For example, set the FortiGate 1 loopback to 10.0.0.1 and the FortiGate 2 loopback to 10.0.0.2.

Configuring OSPF area and interfaces - web-based manager

1. On FortiGate_2, go to **Network > OSPF**.
2. Complete the following.

Router ID	10.0.0.2
Areas	Select **Create New**, enter the **Area** and **Type** and then select **OK**.
Area	0.0.0.0
Type	Regular
Interfaces	
Name	Enter a name for the OSPF interface, ospf_wan1 for example.
Interface	Select the virtual IPsec interface, tunnel_wan1.
IP	0.0.0.0

3. For **Networks**, select **Create New**.
4. Enter the following information for the loopback interface:

IP/Netmask	10.0.0.2/255.255.255.255
Area	0.0.0.0

5. For **Networks**, select **Create New**.
6. Enter the following information for the tunnel interface:

IP/Netmask	10.1.1.0/255.255.255.255
Area	0.0.0.0

7. For **Networks**, select **Create New**.
8. Enter the following information for the local LAN interface:

IP/Netmask	10.31.101.0/255.255.255.255
Area	0.0.0.0

9. Select **Apply**.

Configuring OSPF area and interfaces - CLI

If for example, your loopback interface is 10.0.0.2, your tunnel ends are on the 10.1.1.0/24 network, your local LAN is 10.31.101.0/24, and your virtual IPsec interface is named tunnel_wan1, you would enter:

```
config router ospf
   set router-id 10.0.0.2
   config area
      edit 0.0.0.0
   end
   config network
      edit 1
         set prefix 10.1.1.0 255.255.255.0
      next
```

```
      edit 2
         set prefix 10.31.101.0 255.255.255.0
      next
      edit 2
         set prefix 10.0.0.2 255.255.255.255
      end
   config ospf-interface
      edit ospf_wan1
         set interface tunnel_wan1
         set network-type point-to-point
      end
end
```

Creating a redundant configuration

You can improve the reliability of the OSPF over IPsec configuration described in the previous section by adding a second IPsec tunnel to use if the default one goes down. Redundancy in this case is not controlled by the IPsec VPN configuration but by the OSPF routing protocol.

To do this you:

- Create a second route-based IPsec tunnel on a different interface and define tunnel end addresses for it.
- Add the tunnel network as part of the OSPF network and define the virtual IPsec interface as an additional OSPF interface.
- Set the OSPF cost for the added OSPF interface to be significantly higher than the cost of the default route.

Adding the second IPsec tunnel

The configuration is the same as in Configuring the IPsec VPN on page 197, but the interface and addresses will be different. Ideally, the network interface you use is connected to a different Internet service provider for added redundancy.

When adding the second tunnel to the OSPF network, choose another unused subnet for the tunnel ends, 10.1.2.1 and 10.1.2.2 for example.

Adding the OSPF interface

OSPF uses the metric called cost when determining the best route, with lower costs being preferred. Up to now in this example, only the default cost of 10 has been used. Cost can be set only in the CLI.

The new IPsec tunnel will have its OSPF cost set higher than that of the default tunnel to ensure that it is only used if the first tunnel goes down. The new tunnel could be set to a cost of 200 compared to the default cost is 10. Such a large difference in cost will ensure this new tunnel will only be used as a last resort.

If the new tunnel is called tunnel_wan2, you would enter the following on both FortiGate units:

```
config router ospf
   config ospf-interface
      edit ospf_wan2
         set cost 200
         set interface tunnel_wan2
         set network-type point-to-point
      end
```

```
        end
```

Redundant OSPF routing over IPsec

This example sets up redundant secure communication between two remote networks using an Open Shortest Path First (OSPF) VPN connection. In this example, the HQ FortiGate unit will be called FortiGate 1 and the Branch FortiGate unit will be called FortiGate 2.

The steps include:

1. Creating redundant IPsec tunnels on FortiGate 1.
2. Configuring IP addresses and OSPF on FortiGate 1.
3. Configuring firewall addresses on FortiGate 1.
4. Configuring security policies on FortiGate 1.
5. Creating redundant IPsec tunnels for FortiGate 2.
6. Configuring IP addresses and OSPF on FortiGate 2.
7. Configuring firewall addresses on FortiGate 2.
8. Configuring security policies on FortiGate 2.

Creating redundant IPsec tunnels on FortiGate 1

1. Go to **VPN > IPsec Tunnels**.
2. Select **Create New**, name the primary tunnel and select **Custom VPN Tunnel (No Template)**.
3. Set the following:

Remote Gateway	Static IP Address
IP Address	FortiGate 2's wan1 IP
Local Interface	wan1 (the primary Internet-facing interface)
Pre-shared Key	Enter

4. Go to **VPN > IPsec Tunnels**.
5. Select **Create New**, name the secondary tunnel and select **Custom VPN Tunnel (No Template)**.
6. Set the following:

Remote Gateway	Static IP Address
IP Address	FortiGate 2's wan2 IP
Local Interface	wan2 (the secondary Internet-facing interface)
Pre-shared Key	Enter

Configuring IP addresses and OSPF on FortiGate 1

1. Go to **Network > Interfaces**.
2. Select the arrow for **wan1** to expand the list.

3. Edit the primary tunnel interface and create IP addresses.

IP	10.1.1.1
Remote IP	10.1.1.2

4. Select the arrow for **wan2** to expand the list.
5. Edit the secondary tunnel interface and create IP addresses.

IP	10.2.1.1
Remote IP	10.2.1.2

6. Go to **Network > OSPF** and enter the **Router ID** for FortiGate 1.
7. Select **Create New** in the **Area** section.
8. Add the backbone area of **0.0.0.0**.
9. Select **Create New** in the **Networks** section.
10. Create the networks and select **Area 0.0.0.0** for each one.
11. Select **Create New** in the **Interfaces** section.
12. Create primary and secondary tunnel interfaces.
13. Set a **Cost** of **10** for the primary interface and **100** for the secondary interface.

Configuring firewall addresses on FortiGate 1

1. Go to **Policy & Objects > Addresses**.
2. Create/Edit the subnets behind FortiGate 1 and FortiGate 2.
3. Create/Edit the primary and secondary interfaces of FortiGate 2.

Configuring security policies on FortiGate 1

1. Go to **Policy & Objects > IPv4 Policy**.
2. Create the four security policies required for both FortiGate 1's primary and secondary interfaces to connect to FortiGate 2's primary and secondary interfaces.

Creating redundant IPsec tunnels on FortiGate 2

1. Go to **VPN > IPsec Tunnels**.
2. Select **Create New**, name the primary tunnel and select **Custom VPN Tunnel (No Template)**.
3. Set the following:

Remote Gateway	Static IP Address
IP Address	FortiGate 1's wan1 IP
Local Interface	wan1 (the primary Internet-facing interface)
Pre-shared Key	Enter

4. Go to **VPN > IPsec Tunnels**.
5. Select **Create New**, name the secondary tunnel and select **Custom VPN Tunnel (No Template)**.
6. Set the following:

Remote Gateway	Static IP Address
IP Address	FortiGate 1's wan1 IP
Local Interface	wan2 (the secondary Internet-facing interface)
Pre-shared Key	Enter

Configuring IP addresses and OSPF on FortiGate 1

1. Go to **Network > Interfaces**.
2. Select the arrow for **wan1** to expand the list.
3. Edit the primary tunnel interface and create IP addresses.

IP	10.1.1.2
Remote IP	10.1.1.1

4. Select the arrow for **wan2** to expand the list.
5. Edit the secondary tunnel interface and create IP addresses.

IP	10.2.1.2
Remote IP	10.2.1.1

6. Go to **Network > OSPF** and enter the **Router ID** for FortiGate 2.
7. Select **Create New** in the **Area** section.
8. Add the backbone area of **0.0.0.0**.
9. Select **Create New** in the **Networks** section.
10. Create the networks and select **Area 0.0.0.0** for each one.
11. Select **Create New** in the **Interfaces** section.
12. Create primary and secondary tunnel interfaces.
13. Set a **Cost** of **10** for the primary interface and **100** for the secondary interface.

Configuring firewall addresses on FortiGate 2

1. Go to **Policy & Objects > Addresses**.
2. Create/Edit the subnets behind FortiGate 1 and FortiGate 2.
3. Create/Edit the primary and secondary interfaces of FortiGate 2.

Configuring security policies on FortiGate 2

1. Go to **Policy & Objects > IPv4 Policy**.
2. Create the four security policies required for both FortiGate 2's primary and secondary interfaces to connect to FortiGate 1's primary and secondary interfaces.

Results

1. Go to **Monitor > IPsec Monitor** to verify the statuses of both the primary and secondary IPsec VPN tunnels on FortiGate 1 and FortiGate 2.
2. Go to **Monitor > Routing Monitor**. Monitor to verify the routing table on FortiGate 1 and FortiGate 2. Type **OSPF** for the **Type** and select **Apply Filter** to verify the OSPF route.
3. Verify that traffic flows via the primary tunnel:
   - From a PC1 set to IP:10.20.1.100 behind FortiGate 1, run a tracert to a PC2 set to IP address 10.21.1.00 behind FortiGate 2 and vise versa.
   - From PC1, you should see that the traffic goes through 10.1.1.2 which is the primary tunnel interface IP set on FortiGate 2.
   - From PC2, you should see the traffic goes through 10.1.1.1 which is the primary tunnel interface IP set on FortiGate 1.
4. The VPN network between the two OSPF networks uses the primary VPN connection. Disconnect the wan1 interface and confirm that the secondary tunnel will be used automatically to maintain a secure connection.
5. Verify the IPsec VPN tunnel statuses on FortiGate 1 and FortiGate 2. Both FortiGates should show that primary tunnel is DOWN and secondary tunnel is UP.
6. Go to **Monitor > IPsec Monitor** to verify the status.
7. Verify the routing table on FortiGate 1 and FortiGate 2.
   The secondary OSPF route (with cost = 100) appears on both FortiGate units.
8. Go to **Monitor > Routing Monitor**. Type **OSPF** for the **Type** and select **Apply Filter** to verify OSPF route.
9. Verify that traffic flows via the secondary tunnel:
   - From a PC1 set to IP:10.20.1.100 behind FortiGate 1, run a tracert to a PC2 set to IP:10.21.1.100 behind FortiGate 2 and vice versa.
   - From PC1, you should see that the traffic goes through 10.2.1.2 which is the secondary tunnel interface IP set on FortiGate 2.
   - From PC2, you should see the traffic goes through 10.2.1.1 which is the secondary tunnel interface IP set on FortiGate 1.

OSPF over dynamic IPsec

The following example shows how to create a dynamic IPsec VPN tunnel that allows OSPF.

Configuring IPsec on FortiGate 1

1. Go to **Dashboard** and enter the **CLI Console** widget
2. Create phase 1:

```
config vpn ipsec phase1-interface
   edit "dial-up"
      set type dynamic
      set interface "wan1"
      set mode-cfg enable
      set proposal 3des-sha1
      set add-route disable
      set ipv4-start-ip 10.10.101.0
      set ipv4-end-ip 10.10.101.255
      set psksecret
   next
end
```

3. Create phase 2:

```
config vpn ipsec phase2-interface
   edit "dial-up-p2"
      set phase1name "dial-up"
      set proposal 3des-sha1 aes128-sha1
   next
end
```

Configuring OSPF on FortiGate 1

1. Go to **Dashboard** and enter the **CLI Console** widget.
2. Create OSPF route.

```
config router ospf
   set router-id 172.20.120.22
      config area
         edit 0.0.0.0
         next
      end
      config network
         edit 1
            set prefix 10.10.101.0 255.255.255.0
         next
      end
      config redistribute "connected"
         set status enable
      end
      config redistribute "static"
         set status enable
      end
end
```

Adding policies on FortiGate 1

1. Go to **Policy & Objects > IPv4 Policy** and create a policy allowing OSPF traffic from **dial-up** to **port5**.
2. Go to **Policy & Objects > IPv4 Policy** and create a policy allowing OSPF traffic from **port5** to **dial-up** interfaces.

Configuring IPsec on FortiGate 2

1. Go to **Dashboard** and enter the **CLI Console** widget
2. Create phase 1:

```
config vpn ipsec phase1-interface
   edit "dial-up-client"
      set interface "wan1"
      set mode-cfg enable
      set proposal 3des-sha1
      set add-route disable
      set remote-gw 172.20.120.22
      set psksecret
   next
end
```

3. Create phase 2:

```
config vpn ipsec phase2-interface
   edit "dial-up-client"
      set phase1name "dial-up-client"
      set proposal 3des-sha1 aes128-sha1
      set auto-negotiate enable
   next
end
```

Configuring OSPF on FortiGate 2

1. Go to **Dashboard** and enter the **CLI Console** widget.
2. Create OSPF route.

```
config router ospf
   set router-id 172.20.120.15
      config area
         edit 0.0.0.0
         next
      end
      config network
         edit 1
            set prefix 10.10.101.0 255.255.255.0
         next
      end
      config redistribute "connected"
         set status enable
      end
      config redistribute "static"
         set status enable
      end
end
```

Adding policies on FortiGate 2

1. Go to **Policy & Objects > IPv4 Policy** and create a policy allowing OSPF traffic from **dial-up-client** to **port5**.
2. Go to **Policy & Objects > IPv4 Policy** and create a policy allowing OSPF traffic from **port5** to **dial-up-client** interfaces.

Verifying the tunnel is up

Go to **Monitor > IPsec Monitor** to verify that the tunnel is **Up**.

Results

1. From FortiGate 1, go to **Monitor > Routing Monitor** and verify that routes from FortiGate 2 were successfully advertised to FortiGate 1 via OSPF.
2. From FortiGate 1, go to **Dashboard**. Enter the CLI Console widget and type this command to verify OSPF neighbors:

```
get router info ospf neighbor

OSPF process 0:
Neighbor      ID Pri State Dead  Time     Address Interface
172.20.120.25 1  Full  /     -   00:00:34 10.10.101.1  dial-up_0
```

3. From FortiGate 2, go to **Monitor > Routing Monitor** and verify that routes from FortiGate 1 were successfully advertised to FortiGate 2 via OSPF.
4. From FortiGate 2, go to **Dashboard**. Enter the CLI Console widget and type this command to verify OSPF neighbors:

```
get router info ospf neighbor

OSPF process 0:
Neighbor      ID Pri State Dead  Time     Address     Interface
172.20.120.22 1  Full  /     -   00:00:30 10.10.101.2  dial-up_client
```

BGP over dynamic IPsec

The following example shows how to create a dynamic IPsec VPN tunnel that allows BGP.

Configuring IPsec on FortiGate 1

1. Go to **Policy & Objects > Addresses** and select create new **Address**.

Name	Remote_loop_int
Type	Subnet
Subnet/IP Range	10.10.10.10
Interface	any

2. Create an **Address Group**.

Group Name	VPN_DST
Show in Address List	enable
Members	Remote_loop_int all

3. Go to **Dashboard** and enter the CLI Console widget.
4. Create phase 1:

```
config vpn ipsec phase1-interface
   edit Dialup
      set type dynamic
      set interface wan1
      set mode aggressive
      set peertype one
      set mode-cfg enable
      set proposal 3des-sha1 aes128-sha1
      set peerid dial
      set assign-ip disable
      set psksecret
   next
end
```

5. Create phase 2:

```
config vpn ipsec phase2-interface
   edit dial_p2
      set phase1name Dialup
      set proposal 3des-sha1 aes128-sha1
      set src-addr-type name
      set dst-addr-type name
      set src-name all
      set dst-name VPN_DST
```

```
    next
end
```

Configuring BGP on FortiGate 1

1. Go to **Network > Interfaces** and create a Loopback interface.
2. Set **IP/Network Mask** to **20.20.20.20/255.255.255.255**.
3. Go to **Dashboard** and enter the CLI Console widget.
4. Create a BGP route.

```
config router bgp
   set as 100
   set router-id 1.1.1.1
      config neighbor
         edit 10.10.10.10
            set ebgp-enforce-multihop enable
            set remote-as 200
            set update-source loop
         next
      end
      config redistribute connected
         set status enable
      end
end
```

Adding policies on FortiGate 1

1. Go to **Policy & Objects > IPv4 Policy** and create a policy allowing BGP traffic from **Dialup** to **loop** interfaces.
2. Go to **Policy & Objects > IPv4 Policy** and create a policy allowing BGP traffic from **loop** to **Dialup** interfaces.

Configuring IPsec on FortiGate 2

1. Go to **Dashboard** and enter the CLI Console widget.
2. Create phase 1:

```
config vpn ipsec phase1-interface
   edit Dialup
      set interface wan1
      set mode aggressive
      set mode-cfg enable
      set proposal 3des-sha1 aes128-sha1
      set localid dial
      set remote-gw 172.20.120.22
      set assign-ip disable
      set psksecret
   next
end
```

3. Create phase 2:

```
config vpn ipsec phase2-interface
   edit dial_p2
      set phase1name Dialup
      set proposal 3des-sha1 aes128-sha1
      set keepalive enable
   next
```

```
end
```

Configuring BGP on FortiGate 2

1. Go to **Network > Interfaces** and create a Loopback interface.
2. Set **IP/Network Mask** to **10.10.10.10/255.255.255.255**.
3. Go to **Dashboard** and enter the **CLI Console** widget.
4. Create a BGP route.

```
config router bgp
   set as 200
   set router-id 1.1.1.2
      config neighbor
         edit 20.20.20.20
            set ebgp-enforce-multihop enable
            set remote-as 100
            set update-source loop
         next
      end
      config redistribute connected
         set status enable
      end
end
```

Adding policies on FortiGate 2

1. Go to **Policy & Objects > IPv4 Policy** and create a policy allowing BGP traffic from **Dialup** to **loop** interfaces.
2. Go to **Policy & Objects > IPv4 Policy** and create a policy allowing BGP traffic from **loop** to **Dialup** interfaces.

Adding a static route on FortiGate 2

Go to **Network > Static Routes** and add a route to the remote Loopback interface via Dialup interface.

Destination IP/Mask	20.20.20.20/255.255.255.255
Device	Dialup
Administrative Distance	10

Verifying the tunnel is up

Go to **Monitor > IPsec Monitor** to verify that the tunnel is **Up**.

Results

1. From FortiGate 1, go to **Monitor > Routing Monitor** and verify that routes from FortiGate 2 were successfully advertised to FortiGate 1 via BGP.
2. From FortiGate 1, go to **Dashboard**.
3. Enter the **CLI Console** widget and type this command to verify BGP neighbors:

```
get router info bgp summary
```

4. From FortiGate 2, go to **Monitor > Routing Monitor** and verify that routes from FortiGate 1 were successfully advertised to FortiGate 2 via BGP.
5. From FortiGate 2, go to **Dashboard**.
6. Enter the **CLI Console** widget and type this command to verify BGP neighbors:

```
get router info bgp summary
```

IPsec Auto-Discovery VPN (ADVPN)

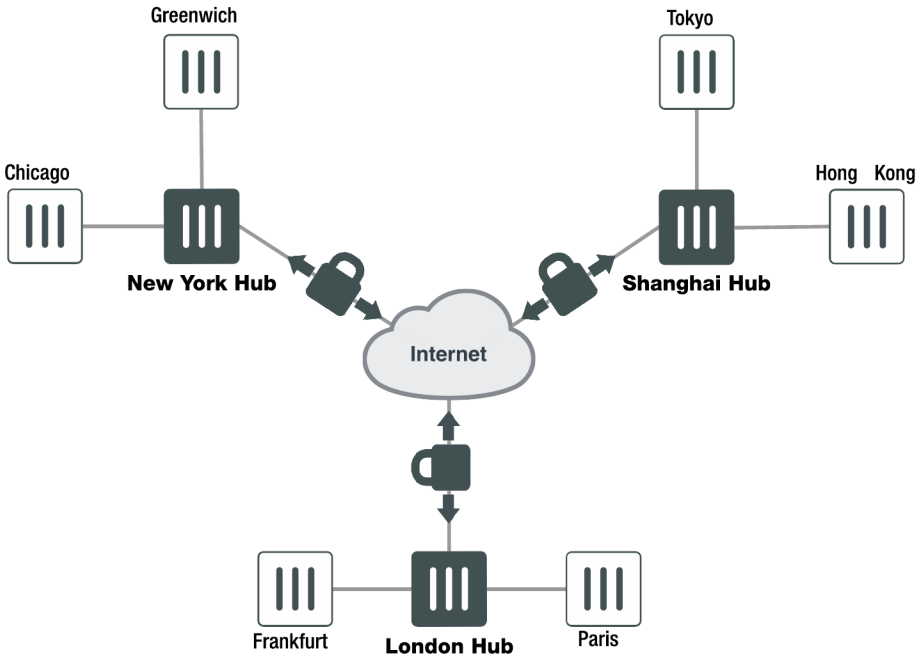
Consider a company that wants to provide direct secure (IPsec) connections between all of its offices in New York, Chicago, Greenwich, London, Paris, Frankfurt, Tokyo, Shanghai, and Hong Kong.

A straightforward solution is to create a full mesh of connections such that every site has eight IPsec configurations, one for each of the other sites. If there were ninety sites, that could still be done but now the configuration is becoming tedious, since every time a new site is added, N-1 other sites have to have their configuration updated.

An efficient and secure alternative is IPsec Auto-Discovery VPN (ADVPN), which allows a minimum amount of configuration per site but still allows direct IPsec connections to be made between every site. RFC 7018 essentially describes this problem, along with some requirements for candidate solutions.

The ADVPN solution involves partitioning the sites into spokes and hubs such that a spoke has to have enough IPsec configuration to enable it to connect to at least one hub. A hub does not have specific configuration for each spoke, so the amount of configuration does not grow with the number of spokes that are connected to that hub. A hub to hub connection would typically involve both hubs having configuration for each other.

So, one possible partition for the original nine sites would be that Chicago and Greenwich would be spokes for the New York hub, Paris and Frankfurt would be spokes for the London hub, and Tokyo and Hong Kong would be spokes for the Shanghai hub:



Once a spoke has established a connection to its hub then initially IPsec traffic to another site transits via one or more hubs. For example, traffic from Chicago to Hong Kong would transit via the New York and Shanghai hubs. This transit traffic then triggers an attempt to create a more direct connection.

In FortiOS:

- Direct connections are only created between the two endpoints that want to exchange traffic (e.g. Chicago and Hong Kong); we do not create intermediate connections (say Chicago to Shanghai, or New York to Hong Kong) as a side-effect.
- Learning the peer subnets is done via a dynamic routing protocol running over the IPsec connections.
- Negotiation of the direct connections is done via IKE.
- Both PSK and certificate authentication is supported.

Example ADVPN configuration

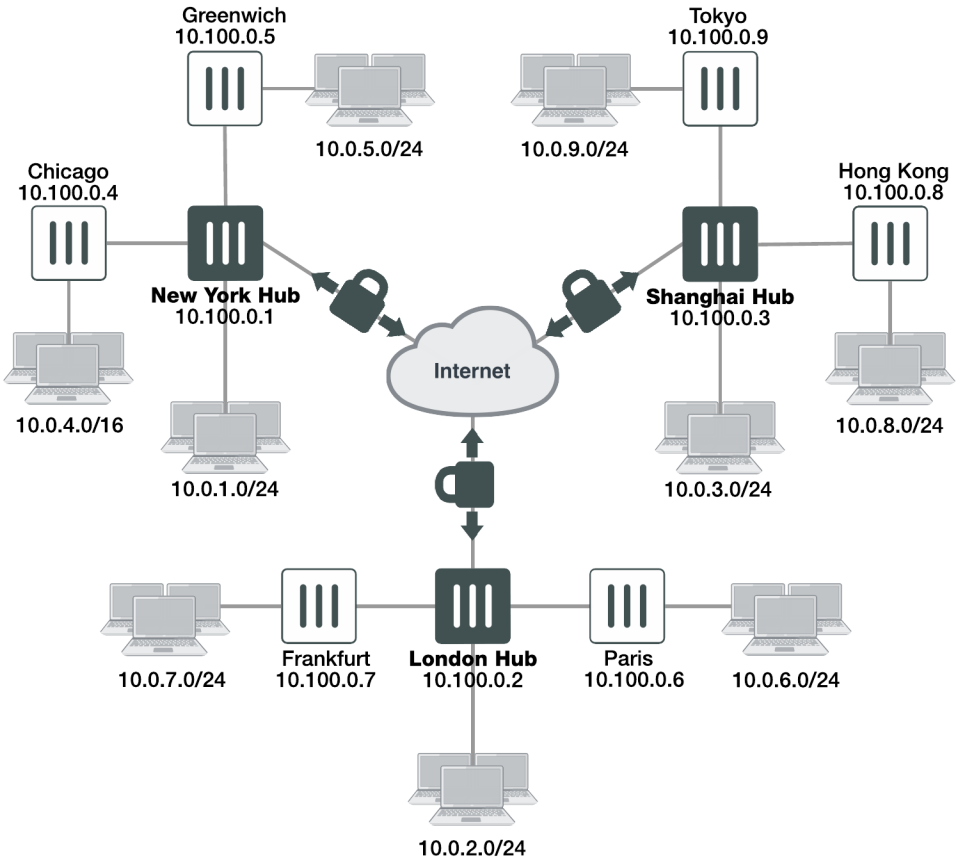
Since dynamic routing with IPsec under FortiOS requires that an interface have an IP address, then for every site a unique IP address from some unused range is allocated. For example we'll assume that 10.100.0.0/16 is unused and so assign the IP addresses:

- Chicago 10.100.0.4
- Greenwich 10.100.0.5
- New York 10.100.0.1
- London 10.100.0.2
- Shanghai 10.100.0.3
- Paris 10.100.0.6
- Frankfurt 10.100.0.7
- Hong Kong 10.100.0.8
- Tokyo 10.100.0.9

We'll assume that each site has one or more subnets that it protects that it wants to make available to the peers. For the purposes of exposition we'll assume there is only one subnet per site and they are allocated as:

- Chicago 10.0.4.0/16
- Greenwich 10.0.5.0/24
- New York 10.0.1.0/24
- London 10.0.2.0/24
- Shanghai 10.0.3.0/24
- Paris 10.0.6.0/24
- Frankfurt 10.0.7.0/24
- Hong Kong 10.0.8.0/24
- Tokyo 10.0.9.0/24

Our example network topology now looks like this:



The configuraton in Chicago would be as follows:

```
config vpn ipsec phase1-interface
   edit "New York"
      set type static
      set interface wan1
      set remote-gw <New-York-IP-address>
      set psk <New-York-PSK>
      set auto-discovery-receiver enable
   next
end
```

The attribute `auto-discovery-receiver` indicates that this IPsec tunnel wishes to participate in an auto-discovery VPN. The IPsec interface would then have its IP assigned according to the Chicago address:

```
config system interface
   edit "New York"
      set ip 10.100.0.4/32
      set remote-ip 10.100.0.1
   next
end
```

RIP (for simplicity, you could use OSPF or BGP) is then configured to run on the IPsec interface and on the Chicago subnet (you could use redistribute connected, but we'll allow for the fact that there may be other subnets learned from another router on the 10.0.4.0/24 subnet):

```
config router rip
```

```
    edit 1
      set prefix 10.100.0.0/16
    next
    edit 2
      set prefix 10.0.4.0/24
    next
  end
```

Other than the firewall policy and a minimal phase 2 configuration, this concludes the configuration for Chicago.

Each spoke would have a similar configuration.

The New York hub would have a dynamic phase 1 for its spoke connections, and two static phase 1s for its connections to the other hubs:

```
config vpn ipsec phase1-interface
  edit "Spokes"
    set type dynamic
    set interface wan1
    set psk <New-York-PSK>
    set auto-discovery-sender enable
    set auto-discovery-psk enable
    set add-route disable
  next
  edit "London"
    set type static
    set interface wan1
    set psk <New-York-London-PSK>
    set auto-discovery-forwarder enable
  next
  edit "Shanghai"
    set type static
    set interface wan1
    set psk <New-York-Shanghai-PSK>
    set auto-discovery-forwarder enable
  next
end
```

The 'Spokes' connection has `set auto-discovery-sender enable` to indicate that when IPsec traffic transits the hub it should optionally generate a message to the initiator of the traffic to indicate that it could perhaps establish a more direct connection. The `set add-route disable` ensures that IKE does not automatically add a route back over the spoke and instead leaves routing to a separately configured routing protocol.

The two inter-hub connections have `set auto-discovery-forwarder enable` to indicate that these connections can participate in the auto-discovery process. The interface IP addresses are assigned:

```
config system interface
  edit "Spokes"
    set ip 10.100.0.1/32
    set remote-ip 10.100.0.254
  next
  edit "London"
    set ip 10.100.0.1/32
    set remote-ip 10.100.0.2
  next
  edit "Shanghai"
    set ip 10.100.0.1/32
```

```
      set remote-ip 10.100.0.3
   next
end
```

Following this, RIP is enabled on the relevant interfaces:

```
config router rip
   edit 1
      set prefix 10.100.0.0/16
   next
   edit 2
      set prefix 10.0.1.0/24
   next
end
```

A similar configuration would be used on the other two hubs.

Traffic flow and tunnel connection

With the configuration in place at all spokes and hubs, assuming all the spokes are connected to a hub, then Chicago would learn (via RIP) that the route to the Hong Kong subnet 10.0.8.0/24 is via its "New York" interface. If a device on the Chicago protected subnet (say 10.0.4.45) attempted to send traffic to the Hong Kong protected subnet (say 10.0.8.13) then it should flow over the New York interface to New York, which should then transmit it over the Shanghai tunnel to Shanghai, which should then send it over the dynamically negotiated Hong Kong tunnel to Hong Kong.

At the point when the traffic transits New York it should notice that the Chicago Spoke tunnel and the Shanghai tunnel have auto-discovery enabled, causing the New York hub to send a message via IKE to Chicago informing it that it may want to try and negotiate a direct connection for traffic from 10.0.4.45 to 10.0.8.13.

On receipt of this message, IKE on Chicago creates the (FortiOS-specific) IKE INFORMATIONAL SHORTCUT-QUERY message which contains the Chicago public IP address, the source IP of the traffic (10.0.4.45), the desired destination IP (10.0.8.13), and the PSK that should be used to secure any direct tunnel (if certificates are confgured, it is assumed that they all share the same CA and so no additional authentication information is required). This message is sent via IKE to New York since routing indicates that New York is the best route to 10.0.8.13.

On receipt of the IKE INFORMATIONAL query, New York checks its routing table to see who owns 10.0.8.13. It finds that 10.0.8.13 should be routed via Shanghai, and since Shanghai is marked as an auto-discovery-forwarder then the query is forwarded.

Shanghai repeats the process, finds that 10.0.8.13 should be routed via its Hong Kong Spoke and so sends it to Hong Kong. Hong Kong checks 10.0.8.13, finds that it owns the subnet, so it remembers the Chicago public IP address (and PSK) and creates an IKE INFORMATIONAL reply message containing its external IP address. To work out where to send the IKE message, the FortiGate does a routing lookup for the original source IP (10.0.4.45), determines that the message should be routed via its Shanghai tunnel and so sends the reply back to Shanghai. The reply then makes its way back to Chicago following the reverse of the path that it used to arrive at Hong Kong.

When the reply makes it back to the Chicago initator then it now knows the IP address of the Hong Kong device. Chicago now creates a new dynamic tunnel with the remote gateway as the Hong Kong public IP address and initiates an IKE negotiation (the dynamic tunnel nameis auto-generated from the tunnel over which it performed the query; in this case it would be called 'New York_0').

This negotiation should succeed since Hong Kong is set up to expect an attempted negotiation from the Chicago public IP address. Once the negotiation succeeds, RIP will start to run on the newly created tunnels at Chicago and Hong Kong. This will update the routing on Chicago (and Hong Kong) so that the prefered route to 10.0.8.0 (10.0.4.0) is via the newly created tunnel rather than via the connection to New York (Shanghai).

Notes about ADVPN in FortiOS

- Auto-discovery is only supported by IKEv1.
- All Spokes must have an IP address that is routable from any other spoke; devices behind NAT are not currently supported.
- The feature requires the use of a dynamic routing protocol. There is no support for IKE handling routing.
- RIP is not a very scalable routing protocol. When there are more than a few spokes it would be advisable to use route summarization to avoid huge RIP updates. Better yet, use BGP instead of RIP.
- It is assumed that spokes will not be used to transit other spoke traffic, for example: traffic from Chicago to Tokyo would not transit an existing Chicago to Hong Kong tunnel even though that has a shorter hop count than a route via New York and Shanghai.
- There is no facility to allow you to filter which traffic that transits the hub should trigger the message sent to the initiator suggesting it create a direct connection. Currently any and all traffic will trigger it.

Logging and monitoring

This section provides some general logging and monitoring procedures for VPNs.

The following topics are included in this section:

Monitoring VPN connections

You can use the monitor to view activity on IPsec VPN tunnels and to start or stop those tunnels. The display provides a list of addresses, proxy IDs, and timeout information for all active tunnels.

Monitoring connections to remote peers

The list of tunnels provides information about VPN connections to remote peers that have static IP addresses or domain names. You can use this list to view status and IP addressing information for each tunnel configuration. You can also start and stop individual tunnels from the list.

To view the list of static-IP and dynamic-DNS tunnels go to **Monitor > IPsec Monitor**.

Monitoring dialup IPsec connections

The list of dialup tunnels provides information about the status of tunnels that have been established for dialup clients. The list displays the IP addresses of dialup clients and the names of all active tunnels. The number of tunnels shown in the list can change as dialup clients connect and disconnect.

To view the list of dialup tunnels go to **Monitor > IPsec Monitor**.

If you take down an active tunnel while a dialup client such as FortiClient is still connected, FortiClient will continue to show the tunnel connected and idle. The dialup client must disconnect before another tunnel can be initiated.

The list of dialup tunnels displays the following statistics:

- The Name column displays the name of the tunnel.
- The meaning of the value in the Remote gateway column changes, depending on the configuration of the network at the far end:
  - When a FortiClient dialup client establishes a tunnel, the Remote gateway column displays either the public IP address and UDP port of the remote host device (on which the FortiClient Endpoint Security application is installed), or if a NAT device exists in front of the remote host, the Remote gateway column displays the public IP address and UDP port of the remote host.
  - When a FortiGate dialup client establishes a tunnel, the Remote gateway column displays the public IP address and UDP port of the FortiGate dialup client.
- The Username column displays the peer ID, certificate name, or XAuth user name of the dialup client (if a peer ID, certificate name, or XAuth user name was assigned to the dialup client for authentication purposes).

- The Timeout column displays the time before the next key exchange. The time is calculated by subtracting the time elapsed since the last key exchange from the keylife.
- The Proxy ID Source column displays the IP addresses of the hosts, servers, or private networks behind the FortiGate unit. A network range may be displayed if the source address in the security encryption policy was expressed as a range of IP addresses.
- The meaning of the value in the Proxy ID Destination column changes, depending on the configuration of the network at the far end:
  - When a FortiClient dialup client establishes a tunnel:
  - If VIP addresses are not used and the remote host connects to the Internet directly, the Proxy ID Destination field displays the public IP address of the Network Interface Card (NIC) in the remote host.
  - If VIP addresses are not used and the remote host is behind a NAT device, the Proxy ID Destination field displays the private IP address of the NIC in the remote host.
  - If VIP addresses were configured (manually or through FortiGate DHCP relay), the Proxy ID Destination field displays either the VIP address belonging to a FortiClient dialup client, or a subnet address from which VIP addresses were assigned.
- When a FortiGate dialup client establishes a tunnel, the Proxy ID Destination field displays the IP address of the remote private network.

VPN event logs

You can configure the FortiGate unit to log VPN events. For IPsec VPNs, Phase 1 and Phase 2 authentication and encryption events are logged. For information about how to interpret log messages, see the FortiGate Log Message Reference.

Logging VPN events

1. Go to **Log & Report > Log Settings**.
2. Verify that the **VPN activity event** option is selected.
3. Select **Apply**.

Viewing event logs

1. Go to **Log & Report > VPN Events**.
2. Select the **Log location**.

Sending tunnel statistics to FortiAnalyzer

By default, logged events include tunnel-up and tunnel-down status events. Other events, by default, will appear in the FortiAnalyzer report as "No Data Available". More accurate results require logs with `action=tunnel-stats`, which is used in generating reports on the FortiAnalyzer (rather than the tunnel-up and tunnel-down event logs). The FortiGate does not, by default, send `tunnel-stats` information.

To allow VPN `tunnel-stats` to be sent to FortiAnalyzer, configure the FortiGate unit as follows using the CLI:

```
config system settings
   set vpn-stats-log ipsec ssl
   set vpn-stats-period 300
end
```

Troubleshooting

This section contains tips to help you with some common challenges of IPsec VPNs.

A VPN connection has multiple stages that can be confirmed to ensure the connection is working properly. It is easiest to see if the final stage is successful first since if it is successful the other stages will be working properly. Otherwise, you will need to work back through the stages to see where the problem is located.

When a VPN connection is properly established, traffic will flow from one end to the other as if both ends were physically in the same place. If you can determine the connection is working properly then any problems are likely problems with your applications.

On some FortiGate units, such as the FortiGate 94D, you cannot ping over the IPsec tunnel without first setting a source-IP. In this scenario, you must assign an IP address to the virtual IPsec VPN interface. Anything sourced from the FortiGate going over the VPN will use this IP address.

If the egress/outgoing interface (determined by kernel route) has an IP address, then use the IP address of the egress/outgoing interface. Otherwise, use the IP address of the first interface from the interface list (that has an IP address).

The first diagnostic command worth running, in any IPsec VPN troubleshooting situation, is the following:

```
diagnose vpn tunnel list
```

This command is very useful for gathering statistical data such as the number of packets encrypted versus decrypted, the number of bytes sent versus received, the SPI identifier, etc. This kind of information in the resulting output can make all the difference in determining the issue with the VPN.

Another appropriate diagnostic command worth trying is:

```
diagnose debug flow
```

This command will inform you of any lack of firewall policy, lack of forwarding route, and of policy ordering issues.

Common IPsec VPN problems

The most common IPsec VPN issues are listed below. Please read thoroughly and note that, although the list is extensive, it is not exhaustive.

This section includes support for the following:

- Failed VPN connection attempts
- Debug output table
- The options to configure policy-based IPsec VPN are unavailable
- The VPN tunnel goes down frequently
- The pre-shared key does not match (PSK mismatch error)
- The SA proposals do not match (SA proposal mismatch)
- Pre-existing IPsec VPN tunnels need to be cleared
- Other potential VPN issues

Failed VPN connection attempts

If your VPN fails to connect, check the following:

- Ensure that the **pre-shared keys** match exactly (see The pre-shared key does not match (PSK mismatch error) below).
- Ensure that both ends use the same P1 and P2 proposal settings (seeThe SA proposals do not match (SA proposal mismatch) below).
- Ensure that you have allowed inbound and outbound traffic for all necessary network services, especially if services such as DNS or DHCP are having problems.
- Check that a static route has been configured properly to allow routing of VPN traffic.

If you are still unable to connect to the VPN tunnel, run the following diagnostic command in the CLI:

```
diagnose debug application ike -1
diagnose debug enable
```

The resulting output may indicate where the problem is occurring. When you are finished, disable the diagnostics by using the following command:

```
diagnose debug reset
diagnose debug disable
```

View the table below for some assistance in analyzing the debug output.

Debug output table

Problem	Debug output	Common causes	Common solutions
Tunnel is not coming up	Error: negotiation failure	IPsec configuration mismatch	Check phase 1 and 2 settings
	Error: no SA proposal chosen	IPsec configuration mismatch	Check phase 1 and 2 settings
	FortiGate using the wrong VPN	Missing or wrong local ID	If there are more than one pre-shared key dial-up VPN with the same local gateway, use aggressive mode and different local IDs
	Error: connection expiring due to XAUTH failure	Wrong username, password, or user group	Check user credentials and user group configuration
	Error: peer has not completed XAUTH exchange	XAuth is disabled in the client	Fix the client's XAuth configuration
Tunnel is bouncing	DPD packets lost	ISP issue	Check the ISP connection

Problem	Debug output	Common causes	Common solutions
Tunnel is up but traffic does not go through	Error: No matching IPsec selector, drop	Quick mode selector mismatch	Fix the quick mode selector
		NAT is enabled	Disable NAT in the firewall policy
	Traffic is not routed to the tunnel	Route or firewall policy misconfiguration	Route-based: traffic must be routed to IPsec virtual interface Policy-based: traffic must match a firewall policy with action set to IPSEC

The options to configure policy-based IPsec VPN are unavailable

Go to **System > Feature Visibility**. Select **Show More** and turn on **Policy-based IPsec VPN**.

The VPN tunnel goes down frequently

If your VPN tunnel goes down often, check the Phase 2 settings and either increase the **Keylife** value or enable **Autokey Keep Alive**.

The pre-shared key does not match (PSK mismatch error)

It is possible to identify a PSK mismatch using the following combination of CLI commands:

```
diag vpn ike log filter name <phase1-name>
diag debug app ike -1
diag debug enable
```

This will provide you with clues as to any PSK or other proposal issues. If it is a PSK mismatch, you should see something similar to the following output:

```
ike 0:TRX:322: PSK auth failed: probable pre-shared key mismatch
ike Negotiate SA Error:
```

The SA proposals do not match (SA proposal mismatch)

The most common problem with IPsec VPN tunnels is a mismatch between the proposals offered between each party. Without a match and proposal agreement, Phase 1 can never establish. Use the following command to show the proposals presented by both parties.

```
diag debug app ike -1
diag debug enable
```

The resulting output should include something similar to the following, where blue represents the remote VPN device, and green represents the local FortiGate.

```
responder received SA_INIT msg
incoming proposal:
proposal id = 1:
   protocol = IKEv2:
      encapsulation = IKEv2/none
         type=ENCR, val=AES_CBC (key_len = 256)
```

```
            type=INTEGR, val=AUTH_HMAC_SHA_96
            type=PRF, val=PRF_HMAC_SHA
            type=DH_GROUP, val=1536.
   proposal id = 2:
      protocol = IKEv2:
         encapsulation = IKEv2/none
            type=ENCR, val=3DES_CBC
            type=INTEGR, val=AUTH_HMAC_SHA_2_256_128
            type=PRF, val=PRF_HMAC_SHA2_256
            type=DH_GROUP, val=1536.
   proposal id = 1:
      protocol = IKEv2:
         encapsulation = IKEv2/none
            type=ENCR, val=AES_CBC (key_len = 128)
            type=INTEGR, val=AUTH_HMAC_SHA_96
            type=PRF, val=PRF_HMAC_SHA
            type=DH_GROUP, val=1536.
```

Pre-existing IPsec VPN tunnels need to be cleared

Should you need to clear an IKE gateway, use the following commands:

```
diagnose vpn ike restart
diagnose vpn ike gateway clear
```

Other potential VPN issues

- Ensure that your FortiGate unit is in NAT/Route mode, rather than Transparent.
- Check your NAT settings, enabling NAT traversal in the Phase 1 configuration while disabling NAT in the security policy. You might need to pin the PAT/NAT session table, or use some of kind of NAT-T keepalive to avoid the expiration of your PAT/NAT translation.
- Ensure that both ends of the VPN tunnel are using Main mode, unless multiple dial-up tunnels are being used.
- Remove any Phase 1 or Phase 2 configurations that are not in use. If a duplicate instance of the VPN tunnel appears on the IPsec Monitor, reboot your FortiGate unit to try and clear the entry.
- If you have multiple dial-up IPsec VPNs, ensure that the peer ID is configured properly on the FortiGate and that clients have specified the correct local ID. Furthermore, in circumstances where multiple remote dialup VPN tunnels exist, each tunnel must have a peer ID set.
- If you are using FortiClient, ensure that your version is compatible with the FortiGate firmware by reading the FortiOS Release Notes.
- If you are using Perfect Forward Secrecy (PFS), ensure that it is used on both peers. You can use the `diagnose vpn tunnel list` command to troubleshoot this.
- Ensure that the **Quick Mode selectors** are correctly configured. If part of the setup currently uses firewall addresses or address groups, try changing it to either specify the IP addresses or use an expanded address range. This is especially useful if the remote endpoint is not a FortiGate device.
- If XAUTH is enabled, ensure that the settings are the same for both ends, and that the FortiGate unit is set to **Enable as Server**.
- Check IPsec VPN Maximum Transmission Unit (MTU) size. A 1500 byte MTU is going to exceed the overhead of the ESP-header, including the additional ip_header,etc. You can use the `diagnose vpn tunnel list` command to troubleshoot this.

- If your FortiGate unit is behind a NAT device, such as a router, configure port forwarding for UDP ports 500 and 4500.

Troubleshooting connection issues

The following section includes troubleshooting suggestions related to:

- LAN interface connection
- Dialup connection
- Troubleshooting VPN connections
- Troubleshooting invalid ESP packets using Wireshark
- Attempting hardware offloading beyond SHA1
- Check Phase 1 proposal settings
- Check your routing
- Try enabling XAuth

LAN interface connection

To confirm whether a VPN connection over LAN interfaces has been configured correctly, issue a ping or traceroute command on the network behind the FortiGate unit to test the connection to a computer on the remote network. If the connection is properly configured, a VPN tunnel will be established automatically when the first data packet destined for the remote network is intercepted by the FortiGate unit.

If the ping or traceroute fail, it indicates a connection problem between the two ends of the tunnel. This may or may not indicate problems with the VPN tunnel. You can confirm this by going to **Monitor > IPsec Monitor** where you will be able to see your connection. A green arrow means the tunnel is up and currently processing traffic. A red arrow means the tunnel is not processing traffic, and this VPN connection has a problem.

If the connection has problems, see Troubleshooting VPN connections on page 227.

Dialup connection

A dialup VPN connection has additional steps. To confirm that a VPN between a local network and a dialup client has been configured correctly, at the dialup client, issue a ping command to test the connection to the local network. The VPN tunnel initializes when the dialup client attempts to connect.

If the ping or traceroute fail, it indicates a connection problem between the two ends of the tunnel. This may or may not indicate problems with the VPN tunnel, or dialup client. As with the LAN connection, confirm the VPN tunnel is established by checking **Monitor > IPsec Monitor**.

Troubleshooting VPN connections

If you have determined that your VPN connection is not working properly through Troubleshooting on page 223, the next step is to verify that you have a phase2 connection.

If traffic is not passing through the FortiGate unit as you expect, ensure the traffic does not contain IPcomp packets (IP protocol 108, RFC 3173). FortiGate units do not allow IPcomp packets, they compress packet payload, preventing it from being scanned.

Testing Phase 1 and 2 connections is a bit more difficult than testing the working VPN. This is because they require diagnose CLI commands. These commands are typically used by Fortinet customer support to discover more information about your FortiGate unit and its current configuration.

Before you begin troubleshooting, you must:

- Configure FortiGate units on both ends for interface VPN
- Record the information in your VPN Phase 1 and Phase 2 configurations - for our example here the remote IP address is 10.11.101.10 and the names of the phases are Phase 1 and Phase 2
- Install a telnet or SSH client such as putty that allows logging of output
- Ensure that the admin interface supports your chosen connection protocol so you can connect to your FortiGate unit admin interface.

For this example, default values were used unless stated otherwise.

Obtaining diagnose information for the VPN connection - CLI

1. Log into the CLI as admin with the output being logged to a file.
2. Stop any diagnose debug sessions that are currently running with the CLI command
   ```
   diagnose debug disable
   ```
3. Clear any existing log-filters by running
   ```
   diagnose vpn ike log-filter clear
   ```
4. Set the log-filter to the IP address of the remote computer (10.11.101.10). This filters out all VPN connections except ones to the IP address we are concerned with. The command is
   ```
   diagnose vpn ike log-filter dst-addr4 10.11.101.10.
   ```
5. Set up the commands to output the VPN handshaking. The commands are:
   ```
   diagnose debug app ike 255
   diagnose debug enable
   ```
6. Have the remote FortiGate initiate the VPN connection in the web-based manager by going to **VPN > IPsec Tunnels** and selecting **Bring up**.

   This makes the remote FortiGate the initiator and the local FortiGate becomes the responder. Establishing the connection in this manner means the local FortiGate will have its configuration information as well as the information the remote computer sends. Having both sets of information locally makes it easier to troubleshoot your VPN connection.
7. Watch the screen for output, and after roughly 15 seconds enter the following CLI command to stop the output.
   ```
   diagnose debug disable
   ```
8. If needed, save the log file of this output to a file on your local computer. Saving the output to a file can make it easier to search for a particular phrase, and is useful for comparisons.

Troubleshooting a Phase 1 VPN connection

Using the output from Obtaining diagnose information for the VPN connection - CLI, search for the word `proposal` in the output. It may occur once indicating a successful connection, or it will occur two or more times for an unsuccessful connection — there will be one proposal listed for each end of the tunnel and each possible

combination in their settings. For example if 10.11.101.10 selected both Diffie-Hellman Groups 1 and 5, that would be at least 2 proposals set.

A successful negotiation proposal will look similar to

```
IPsec SA connect 26 10.12.101.10->10.11.101.10:500
config found
created connection: 0x2f55860 26 10.12.101.10->10.11.101.10:500
IPsec SA connect 26 10.12.101.10->10.11.101.10:500 negotiating
no suitable ISAKMP SA, queuing quick-mode request and initiating ISAKMP SA negotiation
initiator: main mode is sending 1st message...
cookie 3db6afe559e3df0f/0000000000000000
out [encryption]
sent IKE msg (ident-i1send): 10.12.101.10:500->10.11.101.10:500, len=264,
    id=3db6afe559e3df0f/0000000000000000
diaike 0: comes 10.12.101.1:500->10.11.101.1:500,ifindex=26....
```

Note the phrase "`initiator: main mode is sending 1st message...`" which shows you the handshake between the ends of the tunnel is in progress. Initiator shows the remote unit is sending the first message.

Troubleshooting invalid ESP packets using Wireshark

The following section provides information to help debug an encryption key mismatch. The ESP packet invalid error is due to an encryption key mismatch after a VPN tunnel has been established. When an IPsec VPN tunnel is up, but traffic is not able to pass through the tunnel, Wireshark (or an equivalent program) can be used to determine whether there is an encryption mismatch. A mismatch could occur for many reasons, one of the most common is the instability of an ISP link (ADSL, Cable), or it could effectively be any device in the physical connection.

The following information is required to troubleshoot the problem.

- Take a packet sniffer trace on both FortiGates.
- Run the `diag vpn tunnel list` command a few times on both FortiGates when generating traffic that will pass through the tunnel.

In the following example, the error message was seen on the recipient FortiGate:

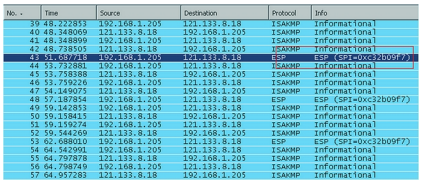
date=2010-12-28 time=18:19:35 devname=Kosad_VPN device_id=FG300B3910600118 log_ id=0101037132 type=event subtype=ipsec pri=critical vd="root" msg="IPsec ESP" action="error" rem_ ip=180.87.33.2 loc_ip=121.133.8.18 rem_port=32528 loc_port=4500 out_intf="port2" cookies="88d40f65d555ccaf/05464e20e4afc835" user="N/A" group="N/A" xauth_user="N/A" xauth_ group="N/A" vpn_tunnel="fortinet_0" status=esp_error error_num=Invalid ESP packet detected (HMAC validation failed). spi=c32b09f7 seq=00000012

This is the output of the command `diag vpn tunnel list` on the FortiGate:

inet ver=1 serial=2 192.168.1.205:4500->121.133.8.18:4500 lgwy=dyn tun=intf mode=auto bound_if=4
proxyid_num=1 child_num=0 refcnt=7 ilast=0 olast=0
stat: rxp=41 txp=56 rxb=4920 txb=3360
dpd: mode=active on=1 idle=5000ms retry=3 count=0 seqno=696
natt: mode=keepalive draft=32 interval=10 remote_port=4500
proxyid=P2_60C_Fortinet proto=0 sa=1 ref=2 auto_negotiate=0 serial=1 src:
0:182.40.101.0/255.255.255.0:0
dst: 0:100.100.100.0/255.255.255.0:0

SA: ref=3 options=0000000d type=00 soft=0 mtu=1428 expire=1106 replaywin=0 seqno=15
life: type=01 bytes=0/0 timeout=1777/1800
dec: spi=29a26eb6 esp=3des key=24 bf25e69df90257f64c55dda4069f01834cd0382fe4866ff2
ah=sha1 key=20 38b2600170585d2dfa646caed5bc86d920aed7ff
enc: spi=c32b09f7 esp=3des key=24 0abd3c70032123c3369a6f225a385d30f0b2fb1cd9687ec8
ah=sha1 key=20 214d8e717306dffceec3760464b6e8edb436c6

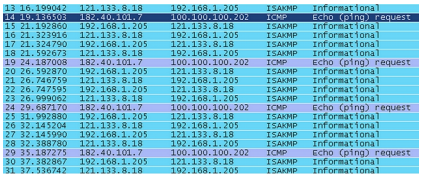
This is the packet capture from the FortiGate:



How to verify if the original packet has been encrypted correctly

To verify, it is necessary to decrypt the ESP packet using Wireshark. Open the packet capture that is taken from initiator FortiGate using Wireshark. Go to **Edit > Preferences**, expand **Protocol** and look for **ESP**. Select "**Attempt to detect/decode encrypted ESP payloads**", and fill in the information for the encryption algorithm and the keys. This information can be obtained from the output of the command `diag vpn tunnel list`.

If the packet was encrypted correctly using the correct key, then the decryption will be successful and it will be possible to see the original package as shown below:



Repeat the decryption process for the packet capture from the recipient firewall. If the decryption failed using the same key, the packet may be corrupted and the interface should then be checked for CRC or packet errors.

Attempting hardware offloading beyond SHA1

If you are trying to off-load VPN processing to a network processing unit (NPU), remember that only SHA1 authentication is supported. For high levels of authentication such as SHA256, SHA384, and SHA512 hardware offloading is not an option—all VPN processing must be done in software—unless using an NP6 (although the NP4lite variation also supports SHA256, SHA384, and SHA512).

Enable/disable IPsec ASIC-offloading

Much like NPU-offload in IKE phase1 configuration, you can enable or disable the usage of ASIC hardware for IPsec Diffie-Hellman key exchange and IPsec ESP traffic. By default hardware offloading is used. For debugging purposes, sometimes it is best for all the traffic to be processed by software.

```
config sys global
   set ipsec-asic-offload [enable | disable]
end
```

Check Phase 1 proposal settings

Ensure that both sides have at least one Phase 1 proposal in common. Otherwise they will not connect. If there are many proposals in the list, this will slow down the negotiating of Phase 1. If its too slow, the connection may timeout before completing. If this happens, try removing some of the unused proposals.

NPU offloading is supported when the local gateway is a loopback interface.

Check your routing

If routing is not properly configured with an entry for the remote end of the VPN tunnel, traffic will not flow properly. You may need static routes on both ends of the tunnel. If routing is the problem, the proposal will likely setup properly but no traffic will flow.

Try enabling XAuth

If one end of an attempted VPN tunnel is using XAuth and the other end is not, the connection attempt will fail. The log messages for the attempted connection will not mention XAuth is the reason, but when connections are failing it is a good idea to ensure both ends have the same XAuth settings. If you do not know the other end's settings enable or disable XAuth on your end to see if that is the problem.

General troubleshooting tips

Most connection failures are due to a configuration mismatch between the FortiGate unit and the remote peer. In general, begin troubleshooting an IPsec VPN connection failure as follows:

1. Ping the remote network or client to verify whether the connection is up. See General troubleshooting tips on page 231.
2. Traceroute the remote network or client. If DNS is working, you can use domain names. Otherwise use IP addresses.
3. Check the routing behind the dialup client. Routing problems may be affecting DHCP. If this appears to be the case, configure a DHCP relay service to enable DHCP requests to be relayed to a DHCP server on or behind the FortiGate server.
4. Verify the configuration of the FortiGate unit and the remote peer. Check the following IPsec parameters:
   - The mode setting for ID protection (main or aggressive) on both VPN peers must be identical.
   - The authentication method (preshared keys or certificates) used by the client must be supported on the FortiGate unit and configured properly.
   - If preshared keys are being used for authentication purposes, both VPN peers must have identical preshared keys.
   - The remote client must have at least one set of Phase 1 encryption, authentication, and Diffie-Hellman settings that match corresponding settings on the FortiGate unit.
   - Both VPN peers must have the same NAT traversal setting (enabled or disabled).
   - The remote client must have at least one set of Phase 2 encryption and authentication algorithm settings that match the corresponding settings on the FortiGate unit.
   - If you are using manual keys to establish a tunnel, the **Remote SPI** setting on the FortiGate unit must be identical to the **Local SPI** setting on the remote peer, and vise versa.

5. To correct the problem, see the following table.

VPN troubleshooting tips

Configuration problem	Correction
Mode settings do not match.	Select complementary mode settings. See Phase 1 parameters on page 46.
Peer ID or certificate name of the remote peer or dialup client is not recognized by FortiGate VPN server.	Check Phase 1 configuration. Depending on the Remote Gateway and Authentication Method settings, you have a choice of options to authenticate FortiGate dialup clients or VPN peers by ID or certificate name (see Phase 1 parameters on page 46). If you are configuring authentication parameters for FortiClient dialup clients, refer to the Authenticating FortiClient Dialup Clients Technical Note.
Preshared keys do not match.	Reenter the preshared key. See Phase 1 parameters on page 46.
Phase 1 or Phase 2 key exchange proposals are mismatched.	Make sure that both VPN peers have at least one set of proposals in common for each phase. See Phase 1 parameters on page 46 and Phase 2 parameters on page 66.
NAT traversal settings are mismatched.	Select or clear both options as required. See Phase 1 parameters on page 46 and Phase 1 parameters on page 46.

A word about NAT devices

When a device with NAT capabilities is located between two VPN peers or a VPN peer and a dialup client, that device must be NAT traversal (NAT-T) compatible for encrypted traffic to pass through the NAT device. For more information, see Phase 1 parameters on page 46.

Troubleshooting L2TP and IPsec

This section describes some checks and tools you can use to resolve issues with L2TP-over-IPsec VPNs.

This section includes:

- Quick checks
- Mac OS X and L2TP
- Setting up logging
- Using the FortiGate unit debug commands

Quick checks

The table below is a list of common L2TP over IPsec VPN problems and the possible solutions.

Problem	What to check
IPsec tunnel does not come up.	Check the logs to determine whether the failure is in Phase 1 or Phase 2. Check the settings, including encapsulation setting, which must be transport-mode. Check the user password. Confirm that the user is a member of the user group assigned to L2TP. On the Windows PC, check that the IPsec service is running and has not been disabled. See Troubleshooting L2TP and IPsec on page 232.
Tunnel connects, but there is no communication.	Did you create an ACCEPT security policy from the public network to the protected network for the L2TP clients? See Troubleshooting L2TP and IPsec on page 232.

Mac OS X and L2TP

FortiOS allows L2TP connections with empty AVP host names and therefore Mac OS X L2TP connections can connect to the FortiGate.

Prior to FortiOS 4.0 MR3, FortiOS refused L2TP connections with empty AVP host names in compliance with RFC 2661 and RFC 3931.

Setting up logging

L2TP logging must be enabled to record L2TP events. Alert email can be configured to report L2TP errors.

Configuring FortiGate logging for L2TP over IPsec

1. Go to **Log & Report > Log Settings**.
2. Select **Event Log**.
3. Select the **VPN activity event** check box.
4. Select **Apply**.

Viewing FortiGate logs

1. Go to **Log & Report > VPN Events**.
2. Select the **Log location** if required.
3. After each attempt to start the L2TP over IPsec VPN, select **Refresh** to view logged events.

Using the FortiGate unit debug commands

Viewing debug output for IKE and L2TP

1. Start an SSH or Telnet session to your FortiGate unit.
2. Enter the following CLI commands

```
diagnose debug application ike -1
diagnose debug application l2tp -1
diagnose debug enable
```

3. Attempt to use the VPN and note the debug output in the SSH or Telnet session.
4. Enter the following command to reset debug settings to default:

```
diagnose debug reset
```

Using the packet sniffer

1. Start an SSH or Telnet session to your FortiGate unit.
2. Enter the following CLI command

```
diagnose sniffer packet any icmp 4
```

3. Attempt to use the VPN and note the debug output.
4. Enter `Ctrl-C` to end sniffer operation.

Typical L2TP over IPsec session startup log entries - raw format

```
2010-01-11 16:39:58 log_id=0101037127 type=event subtype=ipsec pri=notice vd="root" msg="progress IPsec
Phase 1" action="negotiate" rem_ip=172.20.120.151 loc_ip=172.20.120.141 rem_port=500 loc_port=500 out_
intf="port1" cookies="5f6da1c0e4bbf680/d6a1009eb1dde780" user="N/A" group="N/A" xauth_user="N/A" xauth_
group="N/A" vpn_tunnel="dialup_p1" status=success init=remote mode=main dir=outbound stage=1
role=responder result=OK

2010-01-11 16:39:58 log_id=0101037127 type=event subtype=ipsec pri=notice vd="root" msg="progress IPsec
Phase 1" action="negotiate" rem_ip=172.20.120.151 loc_ip=172.20.120.141 rem_port=500 loc_port=500 out_
intf="port1" cookies="5f6da1c0e4bbf680/d6a1009eb1dde780" user="N/A" group="N/A" xauth_user="N/A" xauth_
group="N/A" vpn_tunnel="dialup_p1" status=success init=remote mode=main dir=outbound stage=2
role=responder result=OK

2010-01-11 16:39:58 log_id=0101037127 type=event subtype=ipsec pri=notice vd="root" msg="progress IPsec
Phase 1" action="negotiate" rem_ip=172.20.120.151 loc_ip=172.20.120.141 rem_port=500 loc_port=500 out_
intf="port1" cookies="5f6da1c0e4bbf680/d6a1009eb1dde780" user="N/A" group="N/A" xauth_user="N/A" xauth_
group="N/A" vpn_tunnel="dialup_p1" status=success init=remote mode=main dir=inbound stage=3
role=responder result=DONE

2010-01-11 16:39:58 log_id=0101037127 type=event subtype=ipsec pri=notice vd="root" msg="progress IPsec
Phase 1" action="negotiate" rem_ip=172.20.120.151 loc_ip=172.20.120.141 rem_port=500 loc_port=500 out_
intf="port1" cookies="5f6da1c0e4bbf680/d6a1009eb1dde780" user="N/A" group="N/A" xauth_user="N/A" xauth_
group="N/A" vpn_tunnel="dialup_p1_0" status=success init=remote mode=main dir=outbound stage=3
role=responder result=DONE

2010-01-11 16:39:58 log_id=0101037129 type=event subtype=ipsec pri=notice vd="root" msg="progress IPsec
Phase 2" action="negotiate" rem_ip=172.20.120.151 loc_ip=172.20.120.141 rem_port=500 loc_port=500 out_
intf="port1" cookies="5f6da1c0e4bbf680/d6a1009eb1dde780" user="N/A" group="N/A" xauth_user="N/A" xauth_
group="N/A" vpn_tunnel="dialup_p1_0" status=success init=remote mode=quick dir=outbound stage=1
role=responder result=OK

2010-01-11 16:39:58 log_id=0101037133 type=event subtype=ipsec pri=notice vd="root" msg="install IPsec
SA" action="install_sa" rem_ip=172.20.120.151 loc_ip=172.20.120.141 rem_port=500 loc_port=500 out_
intf="port1" cookies="5f6da1c0e4bbf680/d6a1009eb1dde780" user="N/A" group="N/A" xauth_user="N/A" xauth_
group="N/A" vpn_tunnel="dialup_p1_0" role=responder in_spi=61100fe2 out_spi=bd70fca1

2010-01-11 16:39:58 log_id=0101037139 type=event subtype=ipsec pri=notice vd="root" msg="IPsec Phase 2
status change" action="phase2-up" rem_ip=172.20.120.151 loc_ip=172.20.120.141 rem_port=500 loc_port=500
out_intf="port1" cookies="5f6da1c0e4bbf680/d6a1009eb1dde780" user="N/A" group="N/A" xauth_user="N/A"
xauth_group="N/A" vpn_tunnel="dialup_p1_0" phase2_name=dialup_p2

2010-01-11 16:39:58 log_id=0101037138 type=event subtype=ipsec pri=notice vd="root" msg="IPsec
connection status change" action="tunnel-up" rem_ip=172.20.120.151 loc_ip=172.20.120.141 rem_port=500
loc_port=500 out_intf="port1" cookies="5f6da1c0e4bbf680/d6a1009eb1dde780" user="N/A" group="N/A" xauth_
user="N/A" xauth_group="N/A" vpn_tunnel="dialup_p1_0" tunnel_ip=172.20.120.151 tunnel_id=1552003005
tunnel_type=ipsec duration=0 sent=0 rcvd=0 next_stat=0 tunnel=dialup_p1_0
```

```
2010-01-11 16:39:58 log_id=0101037129 type=event subtype=ipsec pri=notice vd="root" msg="progress IPsec
Phase 2" action="negotiate" rem_ip=172.20.120.151 loc_ip=172.20.120.141 rem_port=500 loc_port=500 out_
intf="port1" cookies="5f6da1c0e4bbf680/d6a1009eb1dde780" user="N/A" group="N/A" xauth_user="N/A" xauth_
group="N/A" vpn_tunnel="dialup_p1_0" status=success init=remote mode=quick dir=inbound stage=2
role=responder result=DONE

2010-01-11 16:39:58 log_id=0101037122 type=event subtype=ipsec pri=notice vd="root" msg="negotiate IPsec
Phase 2" action="negotiate" rem_ip=172.20.120.151 loc_ip=172.20.120.141 rem_port=500 loc_port=500 out_
intf="port1" cookies="5f6da1c0e4bbf680/d6a1009eb1dde780" user="N/A" group="N/A" xauth_user="N/A" xauth_
group="N/A" vpn_tunnel="dialup_p1_0" status=success role=responder esp_transform=ESP_3DES esp_auth=HMAC_
SHA1

2010-01-11 16:39:58 log_id=0103031008 type=event subtype=ppp vd=root pri=information action=connect
status=success msg="Client 172.20.120.151 control connection started (id 805), assigned ip 192.168.0.50"

2010-01-11 16:39:58 log_id=0103029013 type=event subtype=ppp vd=root pri=notice pppd is started

2010-01-11 16:39:58 log_id=0103029002 type=event subtype=ppp vd=root pri=notice user="user1"
local=172.20.120.141 remote=172.20.120.151 assigned=192.168.0.50 action=auth_success msg="User 'user1'
using l2tp with authentication protocol MSCHAP_V2, succeeded"

2010-01-11 16:39:58 log_id=0103031101 type=event subtype=ppp vd=root pri=information action=tunnel-up
tunnel_id=1645784497 tunnel_type=l2tp remote_ip=172.20.120.151 tunnel_ip=192.168.0.50 user="user1"
group="L2TPusers" msg="L2TP tunnel established"
```

Troubleshooting GRE over IPsec

This section describes some checks and tools you can use to resolve issues with the GRE-over-IPsec VPN.

Quick checks

Here is a list of common problems and what to verify.

Problem	What to check
No communication with remote network.	Use the `execute ping` command to ping the Cisco device public interface. Use the FortiGate VPN Monitor page to see whether the IPsec tunnel is up or can be brought up.
IPsec tunnel does not come up.	Check the logs to determine whether the failure is in Phase 1 or Phase 2. Check that the encryption and authentication settings match those on the Cisco device. Check the encapsulation setting: tunnel-mode or transport-mode. Both devices must use the same mode.
Tunnel connects, but there is no communication.	Check the security policies. See Troubleshooting GRE over IPsec on page 235. Check routing. See Troubleshooting GRE over IPsec on page 235.

Setting up logging

Configuring FortiGate logging for IPsec

1. Go to **Log & Report > Log Settings**.
2. Select the **Event Logging**.
3. Select **VPN activity event**.
4. Select **Apply**.

Viewing FortiGate logs

1. Go to **Log & Report > VPN Events**.
2. Select the log storage type.
3. Select **Refresh** to view any logged events.

GRE tunnel keepalives

In the event that each GRE tunnel endpoint has keepalive enabled, firewall policies allowing GRE are required in both directions. The policy should be configured as follows (where the IP addresses and interface names are for example purposes only):

```
config firewall policy
   edit < id >
      set srcintf "gre"
      set dstintf "port1"
      set srcaddr "1.1.1.1"
      set dstaddr "2.2.2.2"
      set action accept
      set schedule "always"
      set service "GRE"
   next
end
```

Cisco compatible keep-alive support for GRE

The FortiGate can send a GRE keepalive response to a Cisco device to detect a GRE tunnel. If it fails, it will remove any routes over the GRE interface.

Configuring keepalive query - CLI:

```
config system gre-tunnel
   edit <id>
      set keepalive-interval <value: 0-32767>
      set keepalive-failtimes <value: 1-255>
   next
end
```

GRE tunnel with multicast traffic

If you want multicast traffic to traverse the GRE tunnel, you need to configure a multicast policy as well as enable multicast forwarding.

- To configure a multicast policy, use the `config firewall multicast-policy` command.
- To enable multicast forwarding, use the following commands:

```
config system settings
   set multicast-forward enable
end
```

Using diagnostic commands

There are some diagnostic commands that can provide useful information. When using diagnostic commands, it is best practice that you connect to the CLI using a terminal program, such as puTTY, that allows you to save output to a file. This will allow you to review the data later on at your own speed without worry about missed data as the diag output scrolls by.

Using the packet sniffer - CLI:

1. Enter the following CLI command:

   ```
   diag sniff packet any icmp 4
   ```

2. Ping an address on the network behind the FortiGate unit from the network behind the Cisco router.

   The output will show packets coming in from the GRE interface going out of the interface that connects to the protected network (LAN) and vice versa. For example:

   ```
   114.124303 gre1 in 10.0.1.2 -> 10.11.101.10: icmp: echo request
   114.124367 port2 out 10.0.1.2 -> 10.11.101.10: icmp: echo request
   114.124466 port2 in 10.11.101.10 -> 10.0.1.2: icmp: echo reply
   114.124476 gre1 out 10.11.101.10 -> 10.0.1.2: icmp: echo reply
   ```

3. Enter `CTRL-C` to stop the sniffer.

Viewing debug output for IKE - CLI:

1. Enter the following CLI commands

   ```
   diagnose debug application ike -1
   diagnose debug enable
   ```

2. Attempt to use the VPN or set up the VPN tunnel and note the debug output.
3. Enter CTRL-C to stop the debug output.
4. Enter the following command to reset debug settings to default:

   ```
   diagnose debug reset
   ```